Simulating Noisy Leakage with Bounded Leakage: Simpler, Better, Faster

Julien Béguinot[1], Ananta Mukherjee[2], Maciej Obremski[3], João Ribeiro[4,5], Lawrence Roy[6], François-Xavier Standaert[1], and Daniele Venturi[7]

[1] Crypto Group, ICTEAM Institute, UCLouvain, Belgium
{julien.beguinot,fstandae}@uclouvain.be
[2] Centre for Quantum Technologies, National University of Singapore
a.mukherjee@u.nus.edu
[3] National University of Singapore
obremski.math@gmail.com
[4] Instituto de Telecomunicações, Lisboa, Portugal
[5] Instituto Superior Técnico, Universidade de Lisboa, Portugal
jribeiro@tecnico.ulisboa.pt
[6] IBM Research Zurich, Switzerland
ldr709@gmail.com
[7] Sapienza University of Rome, Italy
venturi@di.uniroma1.it

Abstract. Theoretical treatments of leakage-resilient cryptography typically work under the assumption that the leakage learned by the adversary (e.g., about an n-bit secret key) is arbitrary but *bounded*, in the sense that the leakage is an ℓ-bit string for some threshold ℓ significantly smaller than n. On the other hand, real-world side-channel attacks on physical implementations of cryptographic protocols produce leakage transcripts that are much longer than n. However, unlike the bounded leakage model, these transcripts are inherently *noisy*. We would like to generically claim that cryptographic schemes resilient to bounded leakage are also resilient to realistic noisy leakages. This boils down to showing that noisy leakages can be simulated using only one bounded leakage query. Prior work (EUROCRYPT 2021 and CRYPTO 2024) made important progress on this problem. Yet, barriers to applicability and interpretability remain, such as the need for large noise levels, the difficulty to estimate the necessary parameters of the leakage distributions, undesirable independence assumptions, and inefficient simulation in certain regimes. In this work, we resolve (or make progress towards resolving) these shortcomings:

1. We show that simple modifications to the simulation strategies in prior work simultaneously allow a cheaper computation of simulation parameters and better parameters than previous results.
2. Leveraging the first item, we obtain a reduction whose amount of extra bounded leakage to simulate correlated signals only increases very mildly. This captures the limited incentive for an adversary to oversample a side-channel signal leading to correlated signal, improving previous results treating these samples as independent.
3. We establish a new "bounded leakage vs. simulation efficiency" tradeoff, roughly trading $\mathcal{O}(\Delta)$ bits leaked by the bounded leakage query

for a $\frac{2^{\Delta}}{\mathrm{poly}(\Delta)}$-factor reduction in simulation complexity. This widens the applicability of our results in the context of computational security, as former simulators were only efficient when simulating from $\mathcal{O}(\log \lambda)$ bits of bounded leakage, with λ the security parameter.

Keywords: Leakage-Resilience · Bounded Leakage · Noisy Leakage

1 Introduction

Physical implementations of cryptographic schemes are often vulnerable to *side-channel* attacks that are not covered by classical theoretical models of security. This includes, for example, attacks based on measuring the power consumption or the electromagnetic radiation of a chip, which are cheap to carry out and have led to serious breaks [39]. The devastating effect of side-channel attacks has led to the rise of *leakage-resilient cryptography*, focusing on the design of cryptographic schemes resilient to a wide class of side-channel attacks.

Leakage-resilient cryptography has seen extraordinary developments over the past 25 years. However, there remains a mismatch between its theory and practice. On the one hand, the theory community has mostly focused on the *bounded leakage model*, which assumes that the adversary can learn an arbitrary function of the secret component (say, a secret key) with at most ℓ output bits, for some predefined threshold ℓ. In particular, ℓ must be smaller than the secret key length, as otherwise no security guarantees are achievable. Arguably the main reason behind this choice is that the bounded leakage model allows us to obtain clean proofs of security, even for advanced functionalities. See the survey of Kalai and Reyzin [33] for an excellent account of research in this direction. On the other hand, real-world side-channel attacks do not inherently produce bounded leakage. In fact, side-channel attacks produce long transcripts (much longer than the secret key under attack) that, however, are *noisy*, and so carry only imperfect information about the secret key, i.e., *noisy leakage* [46].

Ideally, we would like to claim that any scheme that is resilient to an appropriate amount of efficient bounded leakage is also resilient to a large family of realistic noisy leakages. Furthermore, we would like there to be simple, transparent methods for practitioners to test whether their noisy leakages fall into this large family. These noisy leakages are estimated from concrete side-channel measurements on a given device, combined with some standard assumptions. And, to top it off, we would like such an "automatic lifting" claim to hold in the realistic scenario where the adversary can learn multiple *correlated* noisy leakages across multiple rounds of computation on the same secret key.

The exploration of these questions was kickstarted by recent works [14, 43] that consider a general simulation-based paradigm from bounded leakage, which captures the claims above. More precisely, for some secret X supported on $\mathcal{X}$ and some leakage $Z = f(X)$ from X, with f a randomized leakage function, the goal is to design a simulator which can only access X through an arbitrary ℓ-bounded leakage query and, based only on this query and the joint distribution

P_{XZ}, produces a simulated leakage $\tilde{Z}$ such that the secret-true leakage joint distribution P_{XZ} and the secret-simulated leakage joint distribution $P_{X\tilde{Z}}$ are close in statistical distance. The main result in [43] is that, if the conditional distributions $P_{Z|X=x}$ have small *t-hockey-stick divergence* with respect to some distribution Q on average over $x \sim P_X$, i.e.,

$$\mathbb{E}_{x\sim P_X}[\mathsf{SD}_t(P_{Z|X=x}; Q)] = \sum_x P_X(x) \sum_z \max(0, P_{Z|X=x}(z) - 2^t Q(z)) \leq \delta, \quad (1)$$

then the leakage Z can be simulated from approximately t bits of bounded leakage to within statistical distance approximately δ. Empirically, they give evidence that concrete leakage models widely used by practitioners (such as "Hamming weight plus Gaussian noise" and its variants) achieve a good tradeoff between t and δ in Equation (1), leading to good "simulation theorems" from bounded leakage for such noisy leakages.[8] Despite conceptually pointing towards an ideal match between theory and practice, these results remain with a number of shortcomings that may imply the need for additional heuristics, limiting their usability and the interpretability of the resulting proof-based security claims:

- *High noise requirements.* Simulating the generalized noisy leakages of [43] with bounded leakage and small errors requires high noise levels, characterized by Signal-to-Noise Ratios (SNRs) below 10^{-2} in the Hamming weight case [38]. In practice, such SNRs are more likely observed under the heuristic assumption that side-channel adversaries are computationally-bounded and only exploit the leakage signal of small (e.g., 8-bit or 32-bit) intermediate computations, so that the remaining (e.g., 120 or 96) bits produce so-called algorithmic noise [51]. Ensuring security without such an assumption (i.e., assuming a 128-bit leakage signal) and for larger SNRs would therefore strengthen the link between theory and practice (i.e., enable to rely only on physical noise) and allow covering more use cases of less noisy leakage functions.
- *Computationally-intensive estimation of parameters.* Despite noisy Hamming weight leakages being a natural first step towards assessing the practicality of the reductions between bounded leakage and noisy leakage, concrete leakage functions typically deviate slightly from such an idealized model. A popular generalization to capture these deviations is to consider *noisy linear leakages*, where the Hamming weight function is replaced by a weighted sum of bits [47]. Unfortunately, computing the parameters of the generalized noisy leakages in [43] becomes inefficient as the cardinality of the leakage function increases. More precisely, in [43] for each value of t, δ is computed as

$$\delta \triangleq \sum_{x\in\mathcal{X}} \mathbb{P}(X = x) \int_{z\in\mathbb{R}^k} \max(0, f_{Z_x}(z) - 2^t f_Z(z))dz. \quad (2)$$

  The direct computation is unfeasible for 128-bit values, unless the leakage function has some special structure (e.g., Hamming weight leakages, which

[8] These empirical results are obtained by choosing $Q = P_Z$.

enable summing over $n+1$ events rather than 2^n). As a result, evaluations for these practically-relevant (linear) leakages were only provided for 8-bit values so far. Furthermore, as a security parameter, δ should typically be less than 2^{-80}. This can make the numerical approximation of (2) very challenging (e.g., if Monte Carlo methods are used). While such an issue could again be solved under additional heuristics or assumptions, models enabling a more efficient estimation of the security parameters would also strengthen the link between theory and practice and allow covering more use cases.
- *Physical independence assumptions.* Concrete implementations give *multiple* leakage samples to the adversaries. To capture this, [43] considers multivariate leakages $Z_1, \ldots, Z_m$, but only obtain simulation theorems from bounded leakage under the assumption that the Z_i's are conditionally independent given the secret X. This assumption is known to be incorrect: signal and noise correlations have been observed in practice and are recognized as difficult to deal with in related contexts like masking [6]. So capturing them in reductions would once more strengthen the link between theory and practice.
- *Inefficient simulator.* For $t \geq 0$, the simulator of [43] runs in time roughly 2^t, assuming oracle access to the joint probability density function P_{XZ} and to i.i.d. samples distributed according to P_Z. The efficiency of the simulator is of fundamental importance since it directly impacts the security claims for (computationally-secure) cryptographic primitives proven in the bounded leakage model (e.g., simulating with complexity 2^{128} makes claims for AES-based designs void). So improving it would be a useful contribution.

Contributions. Given this state of the art, our contributions are as follows:

- **Simpler and better simulation (Section 3).** Our first contribution is a new simulation theorem for noisy leakages from bounded leakage. It attains better parameters and allows a simpler computation of simulation parameters (i.e., the amount of bounded leakage required and the simulation error) than the previous results of [43], for realistic classes of noisy leakages.
- **Sampling from likeliest conditional, maximal leakage (Section 4).** Our second contribution is another simulation theorem which achieves even better still easy to compute parameters at the cost of harder to sample distribution Q. Namely, we choose Q proportional to the likelihest conditional distribution of $Z|X = x$, i.e., $Q(z) \propto \max_x P_{Z|X}(z|x)$. This theorem is very generic and can be expressed nicely in terms of the maximal leakage $\mathcal{L}(X \to Z)$ which is a standard leakage measure from information theory [31].
- **Treating signal correlations (Section 4.5).** A well-known tension between noisy leakage observed in practice and theoretical bounded leakage is that while the first one can be represented using a large amount of data determined by the sampling rate that the adversary uses when measuring, the second one should be at most a few bits. However, in practice, it is expected that increasing the sampling rate beyond some threshold should lead to very correlated signal (and noise), which does not significant increase the leakages' informativeness [3]. Existing reductions are unable to leverage such

correlations. We obtain the first reduction whose amount of extra bounded leakage to simulate correlated signals only increase very mildly.
- **Trading bounded leakage for simulation efficiency (Section 5).** We modify the simulator to exploit several sampling distributions (Q_i) where samples from Q are transformed into samples from a well chosen Q_i using an extra short hint i. This yields a new "bounded leakage vs. simulation efficiency" tradeoff, roughly trading $\mathcal{O}(\Delta)$ bits leaked by the bounded leakage query for a $\frac{2^{\Delta}}{\mathrm{poly}(\Delta)}$-factor reduction in simulation complexity.
- **Applications (Section 6).** We discuss different applications of our results to obtaining cryptographic primitives with computational security under leakage attacks from the memory and/or the computation.

Differences with respect to the Setting of Masking Compilers. Over the last two decades, there have been efforts to bridge the gap between theoretical and practical leakage models in the separate context of masking. We believe that it is interesting to make a parallel and explain the differences compared to our setting. Ishai, Sahai, and Wagner [30] designed a compiler that transforms any stateful arithmetic circuit into an equivalent implementation that is resilient against (random) *probing leakage*, where an adversary learns the value of a given wire with some probability p.[9] This stimulated a line of work aiming to extend this result to resilience in more realistic noisy leakage models where the adversary learns noisy leakage from every wire in the circuit. In particular, the ISW compiler [30] produces masked circuit implementations resilient to various types of noisy leakages, where noise is measured according to different metrics (such as Euclidean norm [46], statistical distance [20, 21, 22], and others [45, 4, 43]).

Our work departs from this line of research in two ways. First, we study leakage-resilience in a general setting, beyond circuit implementations. Second, we connect noisy leakage and *bounded* leakage. The bounded leakage model is fundamentally different from the probing leakage model. As discussed in [14], simulating probing leakage from a secret $X \in \{0,1\}^n$ using bounded leakage from X requires $\approx n$ bits of bounded leakage. It is also not possible to simulate even simple bounded leakage such as $f(X) = X_1$ using probing leakage.

2 Preliminaries

2.1 Notations

Sets may be denoted by uppercase roman letters or uppercase calligraphic letters such as $\mathcal{S}$ and $\mathcal{T}$. The product distribution $P \otimes Q$ of two distributions P, Q is defined by $(P \otimes Q)(x, y) = P(x)Q(y)$. When P admits a probability density function (pdf) with respect to Lesbegue measure, we denote it f_P so that $P(\mathcal{T}) = \int_{\mathcal{T}} f_P(x)dx$. Random variables are denoted by uppercase roman letters such as X, Y, and Z. The probability mass function of a discrete X

[9] There are various flavors of probing leakage essentially equivalent to each other.

is refered to as P_X. The expectation, variance and support are, respectively, $\mathbb{E}[X]$, $\mathrm{Var}(X)$ and $\mathrm{supp}(X)$. $x \sim P$ means that x is sampled according to P. U_n stands for the uniform distribution over $\{0,1\}^n$. We use $\mathcal{N}(0,\Sigma)$ to denote a zero-mean multivariate Gaussian distribution of dimension k with covariance matrix $\Sigma \in \mathbb{R}^{k\times k}$. When $k=1$, $\Sigma = \sigma^2 \in \mathbb{R}$. Φ is the cumulative distributive function of $\mathcal{N}(0,1)$ defined by $\Phi(x) = \int_{-\infty}^{x} \frac{e^{-\frac{u^2}{2}}}{\sqrt{2\pi}} du$. The base-2 logarithm is denoted by log while ln denotes the natural logarithm. $\lceil x \rceil$ and $\lfloor x \rfloor$ denote the ceiling and floor of $x \in \mathbb{R}$. Let $v \in \mathbb{R}^k$ and $p \geq 1$, the p-norm of v is $\|v\|_p = (\sum_{i=1}^{k} |v_i|^p)^{1/p}$.

We model the secret X by a discrete random variable supported on a finite set $\mathcal{X}$ (e.g. $\mathcal{X} = \{0,1\}^n$ for some n). The leakage from X modeled by Z is supported on $\mathbb{R}^k$ for some integer $k \geq 1$. We further assume that for every $x \in \mathcal{X}$, the conditional distribution of $P_{Z|X=x}$ exists and admits a density with respect to the Lesbegue measure. This setting captures most cases of practical interest.

Remark 1 (Concrete cost of oracle accesses). The computational complexity of the simulation is determined assuming oracle (i.e., $\mathcal{O}(1)$) access to certain samples and densities. Things are stated this way to have general statements whose concrete complexity depends on the leakage under consideration (e.g., whether the leakage is efficiently computable from the secret or not). This was also done in prior work (see [43, Remark 2]). Under common assumptions, oracle accesses are cheap to implement in practice, in which case the stated running time with oracle access corresponds to the real running time.

2.2 Statistical Distance and Hockey-Stick Divergences

The t-hockey-stick divergences generalizes statistical distance as follows.

Definition 1 (Hockey-stick divergence). *Given $t \in \mathbb{R}$, the* t-hockey-stick divergence *between distributions P and Q, denoted $\mathsf{SD}_t(P;Q)$, is defined as*

$$\mathsf{SD}_t(P;Q) = \sup_E [P(E) - 2^t Q(E)]$$

where the supremum is taken over all events and the worst-case event corresponds to the set $E = \{x : f_P(x) \geq 2^t f_Q(x)\}$.

When P and Q on $\mathbb{R}^k$ admits pdfs f_P and f_Q with respect to Lesbegue measure, $\mathsf{SD}_t(P;Q)$ can be rewritten as

$$\mathsf{SD}_t(P;Q) = \int_{\mathbb{R}^k} \max(0, f_P(x) - 2^t f_Q(x)) dx.$$

When $t = 0$, the 0-hockey-stick divergenc is the statistical distance: $\mathsf{SD}_0 = \mathsf{SD}$.

2.3 The Leakage Simulation Paradigm

Simulation of one family of leakages by another was introduced in [14]:

Definition 2 (Leakage simulation). *Given two families $\mathcal{F}(X)$ and $\mathcal{G}(X)$ of leakage functions from X, we say that $\mathcal{F}(X)$ is* ε-simulatable from $\mathcal{G}(X)$ *if for all $f \in \mathcal{F}(X)$ there is a (possibly inefficient) randomized algorithm Sim_f such that*

$$\mathsf{SD}\left(P_{(X,Z)}; P_{(X,\mathsf{Sim}_f^{\mathsf{Leak}(X,\cdot)})}\right) \leq \varepsilon, \tag{3}$$

where $Z = f(X)$ and the oracle $\mathsf{Leak}(X,\cdot)$ accepts a single query $g \in \mathcal{G}(X)$ and outputs $g(X)$, and $\mathsf{Sim}_f^{\mathsf{Leak}(X,\cdot)}$ denotes the output of the simulator with access to this oracle. Furthermore, when $\mathcal{G}(X)$ is the family of all ℓ-bounded leakage functions $g : \mathcal{X} \to \{0,1\}^\ell$ and Equation (3) holds, we say that $\mathcal{F}(X)$ is ε-simulatable from ℓ bits of bounded leakage.

Remark 2. We often think of Sim as made of (possibly randomized) sub-routines $(\mathsf{Sim}_1, \mathsf{Sim}_2)$, where Sim_1 is an algorithm taking as input a description of $f \in \mathcal{F}(X)$ and outputting $g \in \mathcal{G}(X)$ with state $\sigma \in \{0,1\}^*$, while Sim_2 is an algorithm taking as input $g(X)$ and σ and outputting the simulated leakage $\tilde{Z}$.

2.4 A Useful Lemma on Rejection Sampling and Applications

The following lemma controls the parameters for rejection sampling according to a target distribution P using i.i.d. samples from a "source" distribution Q. A proof of this lemma is implicit [43] for discrete distributions P and Q, although this lemma was not explored to its full potential there. The lemma below generalizes rejection sampling to the case of continuous distributions P and Q; the proof can be found in the full version [5].

Lemma 1. *Let P and Q be two continuous distributions over $\mathbb{R}^k$ with pdfs f_P and f_Q, respectively, such that*

$$\mathsf{SD}_t(P;Q) = \delta.$$

For an arbitrary integer T, the following algorithm which:

1. *Produces T i.i.d. samples $z_1, \ldots, z_T$ according to Q;*
2. *For $i = 1, \ldots, T-1$, accepts and returns z_i with probability* $\min\left(1, \frac{f_P(z)}{2^t f_Q(z)}\right)$;
3. *If none of $z_1, \ldots, z_{T-1}$ were accepted, outputs z_T;*

has an output distribution that is $\max(\delta, (1-(1-\delta)2^{-t})^T)$*-close to P in SD.*

As a consequence, the following simulation theorem for noisy leakages characterized by hockey-stick divergences is proven in [43].

Theorem 1 ([43, Theorem 4], adapted). *Let Z be a random variables and $\delta \in (0,1)$. Suppose that there exists $t \in \mathbb{R}$ and a probability distribution Q such that for all $x \in \mathcal{X}$,*

$$\mathsf{SD}_t(P_{Z_x}; Q) \leq \delta.$$

Then, Z is $\max(\delta, (1-(1-\delta)2^{-t})^T)$-simulatable from $\ell = \lceil \log_2 T \rceil$ bits of bounded leakage. Furthermore, the simulator runs in time $\Theta(T)$ assuming oracle access to the pdfs of Q and P_{Z_x} for any $x \in \mathrm{supp}(X)$, and oracle access to i.i.d. samples from Q. More generally if $\mathsf{SD}_t(P_{Z_x}; Q) \leq \delta_x$, then it is $\epsilon = \mathbb{E}_x[\epsilon_x] = \mathbb{E}_x[\max(\delta_x, (1-(1-\delta_x)2^{-t})^T)]$-simulatable. Furthermore, if $t \geq 0$, let $\delta = \mathbb{E}_x[\delta_x]$ by convexity we have

$$\mathbb{E}_x[\max(\delta_x, (1-(1-\delta_x)2^{-t})^T)] \leq (1-2^{-t})^T + \delta(1-(1-2^{-t})^T).$$

If $T = 2^t \ln \frac{1}{\alpha}$, we obtain $\varepsilon \leq \alpha + \delta$.

2.5 Dummy Leakage Dimension Can be Removed

The following lemma shows that simulating leakages that do not carry any sensitive information (i.e., are independent of X) is trivial.

Lemma 2 (Removing Dummy Dimensions). *Let $Z = (Z_0, Z_1)$ be a random variable modeling a leakage about X. Suppose that Z_0 is efficiently simulatable up to SD error ε from ℓ bits of bounded leakage from X, and that $P_{Z|X} = P_{Z_0|X} Q_{Z_1}$ where $Q = Q_{Z_1}$ is efficiently samplable by an algorithm Sample_Q. Then, Z is efficiently simulatable up to SD error ε from ℓ bits of bounded leakage from X.*

Proof. By hypothesis, $\exists\, Y \in \{0,1\}^l$ and an efficient algorithm Sim so that

$$X \to \boxed{\text{Bounded Leakage}} \to Y \to \boxed{\text{Sim}} \to Z_0^{\mathsf{Sim}},$$

where

$$\mathsf{SD}(P_{XZ_0} \| P_{XZ_0^{\mathrm{Sim}}}) \leq \varepsilon.$$

By defining $\widetilde{\mathsf{Sim}}(y) = (\mathsf{Sim}(y), \mathrm{Sample}_Q())$, we directly obtain a simulation of of the leakage variable Z with total SD error at most ε. □

3 Simple and Transparent Simulation

In this section, we show that choosing $Q \neq P_Z$ in Theorem 1 uniformly distributed in a parallelotope containing Z with high probability, we obtain simple simulation theorems from bounded leakage with various advantages compared to the results from [43], for a class of noisy leakages that captures a wide range of models considered by practitioners, formalized by the notion of *noisy leakage*:

Definition 3. *Z supported on $\mathbb{R}^k$ is $(\mathcal{R}, M, \delta)$-*noisy leakage from X *if for every $x \in \mathcal{X}$ and $z \in \mathbb{R}^k$ it holds that $f_{Z_x}(z) \leq M$ and $P_{Z_x}(\mathcal{R}) \geq 1 - \delta$.*

We then need the following intermediate lemma to obtain our theorem:

Lemma 3. *Let Z be $(\mathcal{R}, M, \delta)$-noisy leakage from X, where $\mathcal{R} \subseteq \mathbb{R}^k$ has Lebesgue measure $\mu(\mathcal{R}) > 0$. Then, for Q uniform over $\mathcal{R}$, $t = \log(\mu(\mathcal{R}) \cdot M)$, and every $x \in \mathcal{X}$ we have*

$$\mathsf{SD}_t(P_{Z_x}; Q) = P_{Z_x}(\overline{\mathcal{R}}) \leq \delta.$$

Proof. Fix $x \in \mathcal{X}$ and let f_{Z_x} be the pdf of $P_{Z_x} = P_{Z|X=x}$. Since Q is uniform over $\mathcal{R}$, it has pdf $f_Q(z) = \mu(\mathcal{R})^{-1}$ for every $z \in \mathcal{R}$, and $f_Q(z) = 0$ for $z \notin \mathcal{R}$. Therefore, for every $z \in \mathcal{R}$ we have

$$f_{Z_x}(z) - 2^t Q(z) = f_{Z_x}(z) - \mu(\mathcal{R}) \cdot M \cdot \mu(\mathcal{R})^{-1} = f_{Z_x}(z) - M \leq 0,$$

by definition of M. Consequently, we have

$$\mathsf{SD}_t(P_{Z_x}; Q) = \int_{\mathbb{R}^k} \max(0, f_{Z_x}(z) - 2^t f_Q(z))dz = \int_{\overline{\mathcal{R}}} f_{Z_x}(z)dz = P_{Z_x}(\overline{\mathcal{R}}).$$

by definition of $(\mathcal{R}, M, \delta)$-noisy leakage. □

Combining Lemma 3 with Theorem 1 immediately leads to the following.

Theorem 2. *Let Z be $(\mathcal{R}, M, \delta)$-noisy leakage from X. Then, Z is $\max(\delta, (1-(1-\delta)2^{-t})^T)$-simulatable from $\ell = \lceil \log_2 T \rceil$ bits of bounded leakage from X where $t = \log_2(\mu(\mathcal{R})M)$. Furthermore, the simulator runs in time $\mathcal{O}(T)$ assuming oracle access to the pdfs of the uniform distribution over $\mathcal{R}$ and P_{Z_x} for every $x \in \mathcal{X}$, and to i.i.d. samples according to the uniform distribution over $\mathcal{R}$.*

For a discussion on the complexity of oracle accesses, see Remark 1.

Simpler and Better Simulation Some remarks are due about Theorem 2. First, its proof is very simple, on top of the framework from [43]. Second, its simulation parameters depend only on the Lebesgue measure of the region $\mathcal{R}$ and an upper bound on the pdf of the conditional distributions Z_x. These parameters are easy to compute for realistic noisy leakages. For example, under the widely used "additive noise" assumption that $X \in \{0,1\}^n$ and the leakage Z satisfies $Z = d(X) + \mathcal{N}$ with $d : \{0,1\}^n \to \mathbb{R}$ a deterministic function (a popular choice being $d(X) = \sum_{i=1}^n c_i X_i$ for some real-valued coefficients c_i) and $\mathcal{N}$ an independent real-valued noise with fast-decaying tails (a popular choice being Gaussian noise), then (1) we can take $\mathcal{R}$ to be an interval in $\mathbb{R}$, and (2) $\sup_z f_{Z_x}(z) \leq \sup_z f_{\mathcal{N}}(z)$ for all $x \in \{0,1\}^n$. Concretely, this means that $\mu(\mathcal{R})$ is just the length of an interval in $\mathcal{R}$, and upper bounding the pdfs of the conditional distributions Z_x is equivalent to upper bounding the pdf of $\mathcal{N}$. Furthermore, oracle accesses in Theorem 2 are cheap for practically-relevant noisy (e.g., Hamming weight or linear) leakages and simple regions $\mathcal{R}$.

3.1 Additive Gaussian Noise Leakage

In this section, we discuss an instantiation of Theorem 2 to the special and practically-relevant case of leakages with additive Gaussian noise. As in [4, Section 3.3] we use a whitening transform in order to treat this case.

Lemma 4. *Consider leakages of the form*

$$Z = d(X) + N,$$

where $N \sim \mathcal{N}(0, \Sigma)$. Let W be a whitening transformation (e.g., $W = \Sigma^{-\frac{1}{2}}$) and let $\tilde{Z} = WZ = (W \cdot d)(X) + \tilde{N}$, where $\tilde{N} = WN \sim \mathcal{N}(0, I_k)$. Then, Z can be efficiently simulated from ℓ bits of bounded leakage from X if and only if $\tilde{Z}$ can be efficiently be simulated from ℓ bits of bounded leakage from X.

Proof. Let S be the output of the simulation of Z. Define $\tilde{S} = WS$ to be the output of the simulation of $\tilde{Z} = WZ$. Both simulations have the same simulation error with respect to statistical distance. Therefore, we have $\mathsf{SD}(Z; S) = \sup_E \mathbb{P}(Z \in E) - \mathbb{P}(S \in E) = \sup_E \mathbb{P}(WZ \in WE) - \mathbb{P}(WS \in WE) = \sup_E \mathbb{P}(\tilde{Z} \in WE) - \mathbb{P}(\tilde{S} \in WE) = \sup_{\tilde{E}=WE} \mathbb{P}(\tilde{Z} \in \tilde{E}) - \mathbb{P}(\tilde{S} \in \tilde{E}) = \mathsf{SD}(\tilde{Z}; \tilde{S})$.

Hence, without loss of generality, we may always assume that $\Sigma = I_k$ in the proofs. Furthermore, if the noise is white (i.e., $\Sigma = I_k$) then any orthogonal transformation of the leakage leaves the noise white.

Lemma 5. *Let $N \sim \mathcal{N}(0, I_k)$ and $O \in \mathcal{O}_k(\mathbb{R})$, then $ON \sim \mathcal{N}(0, I_k)$.*

Proof. The covariance matrix associated to ON is $O^\top I_k O = O^\top O = I_k$. □

3.2 Axis-Aligned Parallelotope and Orthonormal Transformation

Let $Z = d(X) + N$, where $N \sim \mathcal{N}(0, I_k)$. Since by Lemma 5 any orthonormal transformation O leaves the noise white, we can transform Z to $OZ = (O \cdot d)(X) + ON$ in order to find the smallest possible axis-aligned parellelotope containing $(O \cdot d)(\mathcal{X})$, $\mathcal{R}_0 = \{(r_1, \ldots, r_k) \mid l_1 \leq r_1 \leq u_1, \ldots, l_k \leq r_k \leq u_k\}$. Up to a translation by $(u_1, \ldots, u_k)$, we may assume that the parrallelotope is anchored at the origin

$$\mathcal{R}_0 = \{(r_1, \ldots, r_k) \mid 0 \leq r_1 \leq w_1, \ldots, 0 \leq r_k \leq w_k\},$$

where $w_j \triangleq u_j - l_j$ is the width of the parallelotope on the j-th dimension. Let $\mathcal{J} \triangleq \{j \mid w_j = 0\} \subseteq \{1, \ldots, k\}$, then $Z_{|\mathcal{J}} = N_{|\mathcal{J}}$ is independent of X. As a consequence, Lemma 2 allows us to simulate Z only by simulating $Z_e = Z_{\mathcal{J}^c}$. Without loss of generality we may assume that $Z_e = (Z_1, \ldots, Z_{k_d})$, where $k_d \leq k$. Hence, we are interested in the parallelotope

$$\mathcal{R}_0^e = \{(r_1, \ldots, r_{k_d}) \mid 0 \leq r_1 \leq w_1, \ldots, 0 \leq r_{k_d} \leq w_{k_d}, \}$$

with strictly positive widths.

Minimum Volume Enclosing Box Finding the orthonormal transformation O that minimizes the volume of the smallest axes-aligned enclosing box of a set of points in $\mathbb{R}^k$ is known as the *minimum volume enclosing box problem* in computational geometry. When $k = 2, 3$ the rotating calliper [48, 50] algorithm provides an efficient and optimal solution. When $k > 3$ a computationally efficient suboptimal solution is to take O as the PCA of the set of points [3]. In particular, if $d(X)$ is supported on a subspace of $\mathbb{R}^k$ of dimension k_d, the PCA will effectively identify a rotation such that $w_i > 0$ for $i \leq k_d$ and $w_i = 0$ otherwise.

3.3 Augmenting the Parallelotope

In order to contain the leakages with high probability and given some $\delta \in (0,1)$, we search $\mathcal{R}_1^e \supseteq \mathcal{R}_0^e$ such that $\mathbb{P}(Z_e \in \mathcal{R}_1^e) \geq 1-\delta$:

$$\mathcal{R}_1^e \triangleq \{(r_1, \ldots, r_{k_d}) \mid -n_1 \leq r_1 \leq w_1 + n_1, \ldots, -n_{k_d} \leq r_{k_d} \leq w_{k_d} + n_{k_d}\}.$$

Since $Z_e = d_e(X) + N_e$ where $N_e \sim \mathcal{N}(0, I_{k_d})$,

$$\mathbb{P}(Z_e \in \mathcal{R}_1^e) \geq \prod_{i=1}^{k_d} (\Phi(n_i) + \Phi(n_i + w_i) - 1).$$

Choosing the n_i constant equal to n, it simplifies to find the smallest n such that

$$\prod_{i=1}^{k_d} (\Phi(n) + \Phi(n + w_i) - 1) \geq 1 - \delta.$$

We cannot find a simple closed form expression for n since it depends on all the w_i. However, it is very easy to find a numerical solution to this problem (e.g., via dichotomy) since the expression is monotonically increasing with n. In the sequel, let $\gamma(\mathbf{w}, \delta)$ be the solution to this optimization problem. Observing that $\Phi(n + w_i) \geq \Phi(n)$, we obtain the following upper bound on γ:

$$\gamma(\mathbf{w}, \delta) \in \left[0, \Phi^{-1}\left(\frac{1}{2}\left(1 + (1-\delta)^{\frac{1}{k_d}}\right)\right)\right].$$

3.4 Main Results using the Parallelotope

Let $Z = d(X) + N$ where $N \sim \mathcal{N}(0, \Sigma)$. Let W be a given invertible whitening transformation of the noise (e.g., $W = \Sigma^{-\frac{1}{2}}$). Let $\mathbf{w} = (w_1, \ldots, w_k)$ be the width of a parallelotope of dimension k that contains $(W \cdot d)(X)$. Without loss of generality, we can assume that $w_1 \geq w_2 \geq \ldots \geq w_k$ and let $k_d \triangleq \max\{i \mid w_i > 0\}$. We therefore obtain the following proposition:

Proposition 1. *Let $\delta > 0$, Z_e is $(\mathcal{R}, M = (2\pi)^{-\frac{k_d}{2}}, \delta)$-noisy from X where $\mathcal{R}$ is a k_d-dimensional parallelotope with widths $w_i + 2\gamma(\mathbf{w}, \delta)$.*

Proof. By construction, we already have constructed $\mathcal{R}$ such that, $\mathbb{P}(Z_e \notin \mathcal{R}) \leq \delta$. Furthermore, since $Z_e = \tilde{d}(X) + N_e$, where $N_e \sim \mathcal{N}(0, I_{k_d})$, it follows that for all $x \in \mathcal{X}$, and for all $z \in \mathbb{R}^{k_d}$,

$$f_{Z_e|X=x}(z) = (2\pi)^{-\frac{k_d}{2}} \exp\left(-\frac{1}{2}\|z - d(x)\|_2^2\right) \leq (2\pi)^{-\frac{k_d}{2}} = M.$$

Theorem 3. *Let*

$$f_{\mathbf{w}}(\delta) \triangleq \frac{(1-\delta)(2\pi)^{\frac{k_d}{2}}}{\prod_{i=1}^{k_d}(w_i + 2\gamma(\mathbf{w}, \delta))}.$$

Then, Z is ε-simulatable from $\ell = \lceil \log_2 T \rceil$ bits of bounded leakage from X where

$$T = \inf_{\delta \in (0,\varepsilon]} \left\lceil (\ln \frac{1}{\varepsilon})(-\ln(1 - f_{\mathbf{w}}(\delta)))^{-1} \right\rceil \leq \left\lceil (\ln \frac{1}{\varepsilon})(-\ln(1 - f_{\mathbf{w}}(\varepsilon)))^{-1} \right\rceil. \quad (4)$$

The simulator runs in time $\mathcal{O}(T)$ assuming oracle access to $x \mapsto d(x)$,Σ and i.i.d. samples from the normal standard distribution.

Proof. Let $\delta > 0$ and Z_e be defined as above. By Proposition 1, Z_e is $(\mathcal{R}, M = (2\pi)^{-\frac{k_d}{2}}, \delta)$-noisy from X, where $\mathcal{R}$ is a k_d-dimensional parallelotope with widths $w_i + 2\gamma(\mathbf{w}, \delta)$. It follows from Theorem 2 with Q the uniform distribution on $\mathcal{R}$ of volume $\mu(\mathcal{R}) = \prod_{i=1}^{k_d}(w_i + 2\gamma(\mathbf{w}, \delta))$ and $t = \log_2(\mu(\mathcal{R})M)$ that Z_e is $\max(\delta, (1 - (1-\delta)2^{-t})^T)$-simulatable from $\ell = \lceil \log_2 T \rceil$ bits of bounded leakage of X and simulator complexity $\mathcal{O}(T)$. Now, for a targeted simulation error ε, the best simulation parameter $T^*(\varepsilon)$ that we achieve is

$$\begin{aligned}
T^*(\varepsilon) &= \inf\{T \mid \exists \delta > 0, \max(\delta, (1 - (1-\delta)2^{-t})^T) \leq \varepsilon\}, \\
&= \inf\{T \mid \exists \delta \in (0, \varepsilon], (1 - (1-\delta)2^{-t})^T \leq \varepsilon\}, \\
&= \inf_{\delta \in (0,\varepsilon]} \left\lceil (\ln \frac{1}{\varepsilon})(-\ln(1 - (1-\delta)2^{-t}))^{-1} \right\rceil, \\
&= \inf_{\delta \in (0,\varepsilon]} \left\lceil (\ln \frac{1}{\varepsilon})(-\ln(1 - f_{\mathbf{w}}(\delta)))^{-1} \right\rceil.
\end{aligned}$$

Lemma 2 ensures that if we can simulate Z_e using ℓ bits of bounded leakage from X then we can simulate Z using ℓ bits of bounded leakage from X. □

As a corollary we obtain in the single-variate case the following expression:

Corollary 1. *Let $Z = d(X) + \sigma\mathcal{N}(0,1)$ and $w = \max_x d(x) - \min_x d(x)$. Then Z is simulatable with statistical error ε from $\ell = \lceil \log_2 T \rceil$ bits of bounded leakage from X where*

$$T = \left\lceil (\ln \frac{1}{\varepsilon})(-\ln\left(1 - \frac{\sqrt{2\pi}(1-\varepsilon)}{\frac{w}{\sigma} + 2\Phi^{-1}(1 - \frac{\varepsilon}{2})}\right))^{-1} \right\rceil.$$

Furthermore, the simulator runs in complexity $\mathcal{O}(T)$ assuming oracle access to the pdfs of Z and Q and to i.i.d. samples from Q.

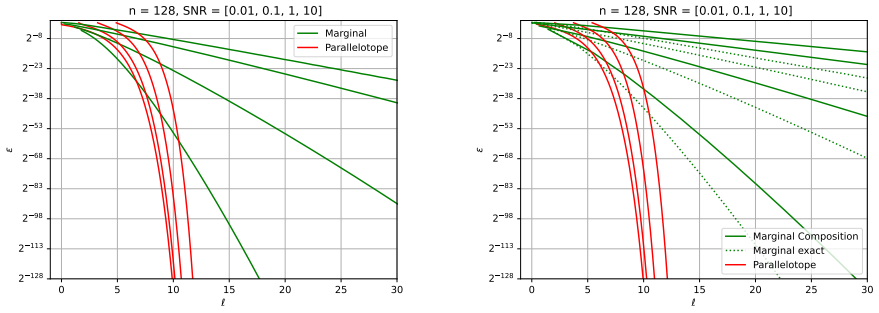
Overcoming limitations of prior works. Theorem 2 and Theorem 3 directly contribute to significantly mitigate two limitations of previous works.

First, and as will be detailed in Section 3.5, we can simulate noisy leakage with bounded leakage in considerably less noisy contexts than [43], to the point that applying Theorem 3 is possible without relying on the algorithmic noise heuristic. This gain derives from our simplified simulation approach, where the amount of bounded leakage needed to simulate mostly depends on the width of the (deterministic part of the) leakage function $d(x)$ (i.e., $\mathbf{w}$ in Theorem 3).

Second, our simplified simulation approach allows a much more efficient estimation of parameters. Namely, the computationally expensive sums/integrals that were limiting the general application of the theorems in [43] to secrets $X \in \{0,1\}^n$ with very small n (or leakage functions with small cardinality) is replaced by the need to characterize the width of the leakage function's deterministic part and the noise level (i.e., a covariance matrix in the Gaussian case). This can be done trivially for the aforementioned linear leakage functions. Remarkably, the ease of computing parameters extends to larger dimensions $k > 1$. In contrast, for the simulation theorems from [43], this computation becomes extremely expensive as k grows, even for very short secrets X.

3.5 Comparison with [43], Hamming Weight Leakage Model

To compare the theorem above with the result from [43], we consider the simple Hamming weight leakage model where $Z = \mathrm{HW}(X) + \sigma N$, $N \sim \mathcal{N}(0,1)$ is an independent additive gaussian noise and X is a uniformly distributed secret in $\{0,1\}^n$. The *signal to noise ratio* (SNR) can be computed as $\mathrm{SNR} = \frac{n}{4\sigma^2}$ in this setting. We obtain the curves in Figure 1a. The approach from [43] only yields better results for large values of ε, which have limited practical interest. By contrast, for ε in the 2^{-128} range, the slow decay of this former work is a limiting factor and implies that simulation requires large ℓ values. Our proposal gets rid of this limitation and succeeds with mild amounts of bounded leakage. As depicted in Figure 1b, the same conclusion holds for bivariate Hamming weight leakages, where the adversary gets $\mathrm{HW}(X)$ twice with independent noise samples.



(a) Univariate Hamming Weight Leakage. (b) Bivariate Hamming Weight Leakage.

Fig. 1: Comparison between our approach and the one from [43] . The number of bit is set to $n = 128$. The red curves correspond to Theorem 3 while the green curve is the main theorem from [43]. For the bivariate case the plain curve is bound using the composition theorem of [43] while the dotted curve is optimal (without composition). We compare the results for SNRs ranging in $\{10^{-2}, 10^{-1}, 1, 10\}$. The lower the SNR, the lower the curve is in the figure.

3.6 Limitations of the Parallelotope

One remaining problem with Theorem 3 is its dependency on the rank of the deterministic part of the leakage model (i.e., k_d). In particular, and while we would expect that if $w_{k_d} \approx 0$, the effect of the corresponding dimensions on the final bound should be mild and vanishing as w_{k_d} approaches zero, this expectation is not consolidated by provable guarantees. This is reflected by the fact that even when $\frac{w}{\sigma} \to 0$, we still have to pay for the $2\Phi^{-1}(1-\frac{\delta}{2})$ term in Corollary 1. Concretely, this problem is due to "jumps" in the parameters each time k_d increases, which is inconvenient since, for leakage models evaluated on real data, it is very likely to obtain a PCA with close to zero but non zero singular values [3]. In the next section we propose another solution which bridges this gap.

4 Beyond Parallelotopes : Maximal Leakages

A concrete consequence of the aforementioned limitations is that increasing the number of dimensions of the leakage traces that have to be simulated unavoidably implies the need of more bounded leakage, as for example captured by the composition theorems in [43]. Unfortunately, in practice, the size of a leakage trace is at least partially under adversarial control. That is, by increasing the sampling rate of an oscilloscope, it is possible to obtain a large number of leakage samples per operation (e.g., a block cipher round in a hardware implementation). Yet, it is also expected that all the leakage samples corresponding to an operation are very correlated. This raises the question whether one can take advantage of such (signal) correlations in order the improve the results of the previous section. In this section, we propose another reduction based on a more complicated sampling distribution Q that solves this problem (and is able to take advantage of the close to zero but non-zero singular values mentioned above). Interestingly, this construction provides reduction parameters which can be expressed in terms of the maximal leakage information measure [31, Definition 1] introduced in information theory. Note that with this construction we obtain a generic theorem and we do not rely on the additive gaussian noise assumption.

4.1 Alternative Sampling Distribution : Likeliest Conditional

We consider the alternative sampling distribution Q defined as follows:

$$Q(z) \triangleq C^{-1} \max_{x \in \mathcal{X}} P_{Z|X=x}(z),$$

where $C \in [1, |\mathcal{X}|]$ is a normalization constant so that Q is a valid distribution. In fact, C is the exponential of the maximal leakage whose definition is recalled below:

$$C \triangleq \int_z \max_{x \in \mathcal{X}} P_{Z|X=x}(z) = 2^{\mathcal{L}(X \to Z)}.$$

Maximal leakage [31, Definition 1] is a standard leakage measure in information theory which belongs to the broader family of Sibson's α-information [49].

Definition 4 (Maximal Leakage). *Consider a given channel $X \to Z$ the maximal leakage provided by Z on X is defined by*

$$\mathcal{L}(X \to Z) \triangleq \log_2 \left(\int_z \max_x p_{Z|X}(z|x) dz \right). \tag{5}$$

Then, by construction, for

$$t = \log_2 C = \mathcal{L}(X \to Z),$$

we have for all x, $\mathsf{SD}_t(P_{Z|X=x}; Q) = 0$. We then obtain the following theorem:

Theorem 4. *Z is ε-simulatable from $\ell = \lceil \log_2 T \rceil$ bits of bounded leakage of X and complexity $\mathcal{O}(T)$ where*

$$T = \left\lceil (\ln \varepsilon)(\ln(1 - 2^{-\mathcal{L}(X \to Z)}))^{-1} \right\rceil.$$

Proof. Z is $\epsilon = \max(0, (1 - (1 - 0)2^{-t})^T) = (1 - 2^{-\mathcal{L}(X \to Z)})^T$-simulatable from $\ell = \lceil \log_2 T \rceil$ bits of bounded leakage of X. The minimum value of T that achieves simulation error ε is hence given by $T = \left\lceil (\ln \varepsilon)(\ln(1 - 2^{-\mathcal{L}(X \to Z)}))^{-1} \right\rceil$. □

Remark 3. Implicitly, this improved sampling strategy has the advantage of grouping the values of x with the leakage channel $P_{Z|X=x}$, which is expected since they are indistinguishable based on the leakage Z.

4.2 Application to Additive Gaussian Noise

In this subsection, we apply Theorem 4 when $Z = d(X) + N$, where $N \sim \mathcal{N}(0, \Sigma)$. The first ingredient is to compute maximal leakages for monovariate additive leakages as it was done for Doeblin coefficient in [4, Proposition 2]:

Lemma 6 (Maximal Leakages for Additive Gaussian Noise). *Assume*

$$Z = d(X) + N,$$

where $N \sim \mathcal{N}(0, 1)$ is an additive gaussian noise. Furthermore, let $\Delta_1, \ldots, \Delta_m$ be the spacing between the points in $d(\mathcal{X})$ sorted by increasing order. Let $\Delta = \sum_{i=1}^m \Delta_i$ be the total the width of the deterministic part of the leakages. Then,

$$\begin{aligned}
\mathcal{L}(X \to Z) &= \log_2 \left(1 + \sum_{i=1}^{m} \left(\Phi(\frac{\Delta_i}{2}) - \Phi(-\frac{\Delta_i}{2}) \right) \right), \\
&\leq \log_2 \left(1 + m \left(\Phi(\frac{\Delta}{2m}) - \Phi(-\frac{\Delta}{2m}) \right) \right), \\
&\leq \log_2 \left(1 + \frac{\Delta}{\sqrt{2\pi}} \right).
\end{aligned}$$

Proof. See the full version [5].

We can then compute easily the maximal leakage in the multivariate case, since it is sub-additive in the following sense [31, Lemma 6]:

Lemma 7 (Sub-additivity). *Let $Z = (Z_1, Z_2)$ and assume that conditionally to X, Z_1 and Z_2 are independent. Then:*

$$\mathcal{L}(X \to Z) \leq \mathcal{L}(X \to Z_1) + \mathcal{L}(X \to Z_2).$$

As a consequence, for additive leakages, the maximal leakage can be easily evaluated using only the widths sides of a parrallelotope supporting the deterministic part of the leakages. It leads to the following theorem.

Theorem 5. *Let $Z = d(X) + N$ where $N \sim \mathcal{N}(0, \Sigma)$. Let W be a given invertible whitening transformation of the noise (e.g. $W = O \cdot \Sigma^{-\frac{1}{2}}$ where O is an orthonormal transformation). Let $\mathbf{w} = (w_1, \ldots, w_k)$ be the width of a k-dimensional parallelotope containing $(W \cdot d)(X)$. Then Z can be simulated from $\ell = \lceil \log_2 T \rceil$ bits of bounded leakage from X with statistical error ε where*

$$T = \left\lceil (\ln \frac{1}{\varepsilon})(-\ln \left(1 - \prod_{i=1}^{k} \frac{1}{1 + \frac{w_i}{\sqrt{2\pi}}}\right))^{-1} \right\rceil.$$

Furthermore, the simulator runs in time $\mathcal{O}(T)$.

Again, the parameters can be easily computed as they depend only on the width of the deterministic part of the leakages. Compared to Theorem 3, the theorem does not depend on the rank k_d which makes its expression more suitable for practical evaluation. Indeed, as w_i approaches zero, $1 + \frac{w_i}{\sqrt{2\pi}}$ approaches 1 so that the contribution of this term smoothly disappears from the theorem. As a particular case, we obtain the following expression in the monovariate case:

Corollary 2. *If $Z = d(X) + \sigma\mathcal{N}(0, 1)$ and $w = \max_x d(x) - \min_x d(x)$, then Z is simulatable with statistical error ε from $\ell = \lceil \log_2 T \rceil$ bits of bounded leakage from X where*

$$T = \left\lceil (\ln \frac{1}{\varepsilon})(\ln(1 + \frac{\sigma\sqrt{2\pi}}{w}))^{-1} \right\rceil.$$

Furthermore, the simulator runs in complexity $\mathcal{O}(T)$ assuming oracle access to the pdfs of Z and Q and to i.i.d. samples from Q.

Overcoming limitations from prior works. Theorem 4 and Theorem 5 solve one of the most important limitations from the state of the art. Indeed, if Z contains many correlated samples, then for an appropriate choice of O (e.g., using Principal Component Analysis [3]), we can ensure that $w_i \approx 0$ for most of the w_i, so that their contributions cancel. This captures (to the best of our knowledge, for the first time) the limited interest for an adversary to oversample her measurements as soon as it mostly adds correlated signal.

4.3 Limitations of Maximal Leakage

The main drawback of the approach described in this section is that for large $|\mathcal{X}|$, it may be hard to sample from the distribution Q. As an example, if $X \in \{0,1\}^n$ and the leakage model d is linear (i.e., there exists $a \in \mathbb{R}^n$ such that $d(x) = \langle a|x\rangle = \sum_{i=1}^n x_i a_i$), then computing $Q(z)$ requires finding $x \in \{0,1\}^n$ which minimizes $|z - \sum_{i=1}^n x_i a_i|$. This problem is a variant of the closest subset sum problem [35] which can be prohibitively expensive for large values of n in general. In particular, the cost of sampling from Q is higher than for the parallelotope where the sampling was very cheap. In the next sub-section, we describe an hybrid approach which efficiently bridges the gap between both approaches.

4.4 Maximal Leakage for Tight Parameters Parallelotope for Efficient Sampling

To motivate this hybrid construction, we start with two observations:

1. The statistical error in the simulator using a parralelotope is due to the tail of P_Z. The widths of the parralelotope therefore need to increase in order to contain P_Z with high probability and simulate accurately.
2. Using maximal leakages, we can avoid this problem but the corresponding distribution may be complicated to compute for large alphabets. However, it is actually easy to compute this distribution when z is not in $d(\mathcal{X})$: it is then sufficient to know the maximal and minimal value of $d(\mathcal{X})$.

For simplicity, we begin with the monovariate case where $Z = d(X) + N(0,1)$. Let $d_{\min} = \min_x d(x)$, $d_{\max} = \max_x d(x)$ and $w = d_{\max} - d_{\min}$. The two observations motivate to define the following sampling distribution:

$$Q(z) = \begin{cases} C_0 p_N(z - d_{\min}), & z < d_{\min}, \\ C_1, & d_{\min} \le z \le d_{\max}, \\ C_0 p_N(z - d_{\max}), & z > d_{\max}, \end{cases}$$

where C_0, C_1 are normalization constants so that Q is a valid probability density function. Now observe that $\int_{z<d_{\min}} p_N(z - d_{\min})dz + \int_{z>d_{\max}} p_N(z - d_{\max})dz = \int_{z<0} p_N(z)dz + \int_{z>0} p_N(z)dz = \int p_N(z)dz = 1$. Then, necessarily, $C_0 + C_1 w = 1$ so that we always have $C_1 = \frac{1-C_0}{w}$. It follows that $\mathsf{SD}_t(P_{Z_x}; Q) = 0$ for all x provided that $2^t C_0 \ge 1$ and $2^t(1 - C_0)/w \ge \frac{1}{\sqrt{2\pi}}$. That is, we need $2^t \ge \max(C_0^{-1}, \frac{w}{\sqrt{2\pi}(1-C_0)})$. The best choice for C_0 equalizes both terms in the maximum, i.e., $C_0 = (1 + \frac{w}{\sqrt{2\pi}})^{-1}$. The corresponding value for t is $\log_2(1 + \frac{w}{\sqrt{2\pi}})$. We therefore obtain a simulation with statistical distance error ε from $\ell = \lceil \log_2 T \rceil$ bits of bounded leakage from X where

$$T = \left\lceil (\ln \frac{1}{\varepsilon})(-\ln\left(1 - \frac{1}{1 + \frac{w}{\sqrt{2\pi}}}\right))^{-1} \right\rceil,$$

which is the same parameter that we obtained by upper bounding the maximal leakage with Lemma 6. We have the following generalization for arbitrary k

Theorem 6. *Let $Z = d(X)+N$ where $N \sim \mathcal{N}(0,\Sigma)$. Let W be a given invertible whitening transformation of the noise (e.g. $W = \Sigma^{-\frac{1}{2}}$). Let $\mathbf{w} = (w_1, \ldots, w_k)$ be the width of a k-dimensional parallelotope containing $(W \cdot d)(X)$. The Z can be simulated from $\ell = \lceil \log_2 T \rceil$ bits of bounded leakage from X with statistical error ε where*

$$T = \left\lceil (\ln \frac{1}{\varepsilon})(-\ln\left(1 - \prod_{i=1}^{k} \frac{1}{1+\frac{w_i}{\sqrt{2\pi}}}\right))^{-1} \right\rceil.$$

Furthermore, the simulator runs in time $\mathcal{O}(T)$.

Proof. Let $Z_e = WZ$. Observe that $P_{Z_e|X=x} = \bigotimes_{i=1}^{k} P_{Z_{e,i}|X=x}$. We apply the same construction to obtain a sampling distribution Q_i for each $Z_{e,i}$ with $t_i = \log_2(1+\frac{w}{\sqrt{2\pi}})$. Let $Q = Q_1 \otimes \ldots \otimes Q_k$ and $t = \sum_{i=1}^{k} t_i$ then, $\mathsf{SD}_t(P_{Z_e|X=x}; Q) \leq 0$. The theorem follows from Theorem 1. □

As a consequence, we obtain similar simulation parameters while using a distribution Q which is simple to sample from (since the probability density function is simple, it can be obtained via inversion of the repartition function).

4.5 Observing Close but not Identical Functions

In this subsection, we first consider simulated multivariate leakages to discuss the impact of noise and signal correlations in a controlled setup (we then validate our method with real multivariate leakage in Section 4.6).

We expect from a satisfying theorem that when an adversary oversamples the leakages (i.e., observes very correlated samples), leading to increasing the size of the measured traces significantly, the amount of bounded leakage needed to simulate should not increase. To discuss that, we model both the leakage function (which differs from the Hamming weight leakage model used in Section 3.5) and the additive noise between two samples as autoregressive processes. This enables us to discuss the impact of signal and noise correlations and their interplay.

Let $\langle \mathbf{a}|X\rangle \triangleq \sum_{i=i}^{n} a_i X_i$. When $\mathbf{a} = (1, \ldots, 1)$ this corresponds for instance to the Hamming weight function. We consider k leakage coefficients $\mathbf{a}_1, \ldots, \mathbf{a}_N$ such that

$$Z = (\langle X \mid \mathbf{a}_1\rangle, \ldots, \langle X \mid \mathbf{a}_N\rangle) + N,$$

where $N \sim \mathcal{N}(0, \Sigma_k)$ and $\Sigma_k = \sigma(\rho^{\frac{|i-j|}{k-1}})_{1\leq i,j\leq k}$ is the Kac-Murdock-Szegö [32] (KMS) matrix of dimension k and parameter $\rho^{\frac{1}{k-1}}$ scaled by σ^2, which would appear if the noise components $N_1, \ldots, N_k$ are obtained from a white noise $\epsilon_1, \ldots, \epsilon_k$ of variance σ^2 via an autoregressive relation of order one defined by $N_1 = \epsilon_1$ and $N_{i+1} = \rho^{\frac{1}{k-1}} N_i + (1-\rho^{\frac{1}{k-1}})^{\frac{1}{2}} \epsilon_{i+1}$. The choice to take $\rho^{\frac{1}{k-1}}$ as the parameter is made so that for all choice of k, $\mathrm{Cov}(N_1, N_k) = \sigma^2\rho$. (When $k = 2$ it reduces to the covariance matrix considered in [4].) We choose the leakage model coefficient as $\mathbf{a}_1 \approx \ldots \approx \mathbf{a}_N \approx (1, \ldots, 1)$ which corresponds to a perturbated Hamming weight leakage model. More precisely, for all $j \in \{1, \ldots, n\}$,

we draw $(\mathbf{a}_{i,j})_{1\leq i\leq k}$ as $(1,\ldots,1)+\mathcal{N}(0,\Sigma_{a,k})$ where $\Sigma_{a,k}=\sigma_a(\rho_a^{\frac{|i-j|}{k-1}})_{1\leq i,j\leq k}$ is another KMS matrix of dimension k. The parameter σ_a controls how far the coefficients are drawn from the Hamming weight leakage model while ρ_a controls the correlation between these modications between two time samples. If $\sigma_a>0$, as k increases the rank k_d of the leakage model (which is the rank of the (k,n) matrix A obtained by stacking the $\mathbf{a}_i$) is quickly equal to the number of bits of X, $(k_d=n)$. Hence, the approach using parallelotope quickly degrades while the maximal leakage approach better captures that observing close but not identical leakages does not help the adversary. This shows the effect of oversampling the measurements to observe increasingly similar leakage functions.

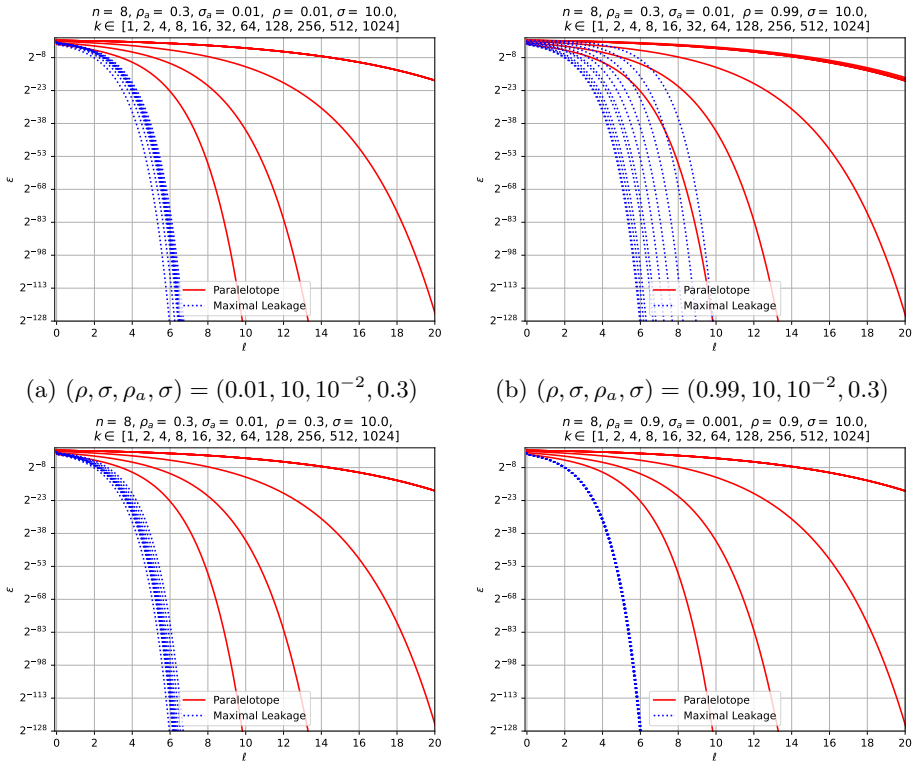
Concretely, we have five parameters in this simulation $(k,\rho,\sigma,\rho_a,\sigma_a)$ As $\sigma_a\to 0$ or $\rho_a\to 1$, the observed signal is more and more correlated for the different samples. As $\rho\to 1$, the observed noise becomes more and more correlated. Overall, we can make the following observations from Figure 2:

- If the signal is very correlated, noise correlation is detrimental to the attacker.
- If the signal is less correlated the noise correlation helps the attacker.
- The attacker's incentive to use more samples (e.g., by increasing the sampling rate of her oscilloscope) decreases with their signal correlation.

4.6 Targeting Real Measurements

In order to confirm the ability of the previous theorems to deal with correlated samples as observed in practice, we finally measured a 10-cycle FPGA implementation of the AES and estimated linear models as prescribed in [47]. We first modeled the leakage of the first S-box in the first cipher round (and clock cycle) for this purpose, for which we measured ten (correlated) samples. For each sample we then estimated the eight (real-valued) weights of a linear model. We additionally estimated the noise covariance matrix for these samples. The maximum noise correlation observed between consecutive samples was estimated at 93%, and dropped at 62% for the most remote samples. The signal correlations turned out to be even higher, with each sample itself strongly correlated with the Hamming weight model. In Figure 3 we apply our theorem for Z with $k=1,\ldots,10$ samples and compare it with the approach of the previous section. Results obtained via maximum leakages almost superimpose, confirming that nearly no additional bounded leakage is needed to simulate (very) correlated samples. By contrast, the approach with the parallelotope requires an amount of bounded leakage that grows roughly linearly with the number of dimensions.

Note that this theorem is particularly useful for dealing with the (possibly many) time samples of a single clock cycle in an implementation, which are expected to be correlated. It is less relevant in the case of samples corresponding to different operations (e.g., two cycles corresponding to two AES rounds) since in this case, the side-channel signal is expected to be uncorrelated.



(a) $(\rho, \sigma, \rho_a, \sigma) = (0.01, 10, 10^{-2}, 0.3)$ (b) $(\rho, \sigma, \rho_a, \sigma) = (0.99, 10, 10^{-2}, 0.3)$

(c) $(\rho, \sigma, \rho_a, \sigma) = (0.3, 10, 10^{-2}, 0.3)$ (d) $(\rho, \sigma, \rho_a, \sigma) = (0.9, 10, 10^{-3}, 0.9)$

Fig. 2: Simulation results for different set of parameters $(\rho, \sigma, \rho_a, \sigma)$ and k ranging from 2^0 to 2^{10}. We consider a single byte $n = 8$ in these experiments.

5 Trading Bounded Leakage for Simulation Efficiency

When dealing with a computational objects we sometimes need the simulator to be efficient, otherwise the mere act of translating the leakage from one family to another would not preserve the security of underlying object. Eventually, the simulator behind Theorem 2 runs in time roughly $\mu(\mathcal{R}) \cdot M \cdot \ln \frac{1}{\varepsilon}$ if Z is $(\mathcal{R}, M, \delta)$-noisy leakage from X. Therefore, this result hits the same efficiency barrier as the results from [43]: the simulation is efficient with respect to some security parameter λ for up to $\mathcal{O}(\log \lambda)$ bits of bounded leakage. While this covers many noisy leakage functions of practical interest, we find it natural to explore *efficient* simulation beyond the settings with $\mathcal{O}(\log \lambda)$ bits of bounded leakage considered so far in the literature.

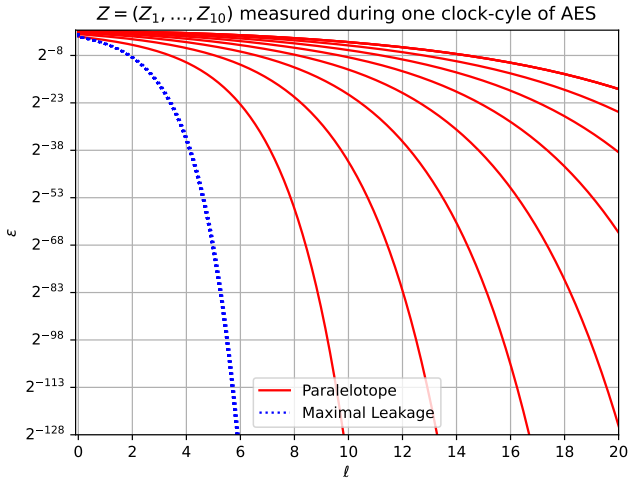


Fig. 3: Simulation results for a 10-sample leakage trace with correlated samples corresponding to one cycle of a 128-bit FPGA implementation of the AES.

5.1 High Level Idea

The original simulation framework in [43] requires that the source distribution Q be independent of X, in the sense that in

$$\mathsf{SD}_t(P_{Z_x}; Q) := \int_{\mathbb{R}^k} \max(0, f_{Z_x}(z) - 2^t f_Q(z))dz \leq \delta_x, \tag{6}$$

the distributions P_{Z_x} are compared to the *same* Q, for all choices of x. We extend the framework of [43] to choose a potentially different source distributions Q_x for each x, which turns out to be the key for the new efficiency vs. bounded leakage tradeoffs. First, recall that the simulator in the framework of [43]:

1. Generates $L = 2^\ell$ i.i.d. samples $z_1, \ldots, z_L$ according to a fixed in advance source distribution Q;
2. Uses $z_1, \ldots, z_L$ to create a bounded leakage query to X that, upon seeing $X = x$ uses $z_1, \ldots, z_L$ to rejection sample from the true leakage distribution P_{Z_x} and returns to the simulator the selected index $i \in [L]$;
3. Outputs the accepted sample z_i as the simulated leakage.

This requires $\log L = \ell$ bits of bounded leakage. The statistical simulation error corresponds to the probability that none of the L samples is accepted.

The bounded leakage query above could, in principle, perform rejection sampling from P_{Z_x} based on a different source distribution Q_x depending on the observed secret x. There is, however, the apparent barrier that the simulator must generate the i.i.d. samples to be used in the rejection sampling procedure *without seeing* $X = x$. We overcome this by using extra bounded leakage in the

bounded leakage query to tell the simulator how to transform a sample from some distribution Q independent of x into a sample from the "correct" distribution Q_x. More precisely, the barrier above disappears if the simulator:

1. Generates L i.i.d. samples $z_1, \ldots, z_L$ according to some fixed in advance distribution Q;
2. The bounded leakage query sees $z_1, \ldots, z_L$ and $X = x$, and transforms $z_1, \ldots, z_L$ into i.i.d. samples $z'_1, \ldots, z'_L$ from a distribution Q_x that *depends on* x. This is possible since the bounded leakage query knows x;
3. The bounded leakage query performs rejection sampling from P_{Z_x} using $z'_1, \ldots, z'_L$.
4. Returns the selected index $i \in [L]$ to the simulator, along with a *short* explanation of how to turn z_i into z'_i;
5. The simulator compute z'_i from z_i using the short explanation, and outputs z'_i as the simulated leakage.

This strategy works so long as we can compute z'_i from z_i and a small amount of bounded leakage. This means that we must choose Q and the tuple of distributions $(Q_x)_{x \in \mathcal{X}}$ above carefully: On the one hand performing rejection sampling based on Q_x should be much cheaper than doing it based on Q to obtain a significant improvement in the efficiency of the simulator. On the other hand, we must be careful about not requiring a lot of information (bounded leakage) to transform a sample from Q into a sample from Q_x.

We now explain how we construct the Q_x's taking inspiration from Riemann integration, where the integral of a function is approximated by the integral of finer and finer step functions. Without loss of generality Z lands in the interval $\mathcal{I} = [0, D]$ with probability at least $1 - \delta$, possibly with extremely large $D = \exp(\lambda)$. Then, for a parameter $d \ll D$, we partition $\mathcal{I}$ into $m = D/d$ disjoint intervals $\mathcal{I}_1, \ldots, \mathcal{I}_m$ each of length d. A first idea for defining Q_x would be to have Q_x first pick the interval $\mathcal{I}_j$ with probability proportional to the likelyhood that $Z_x | Z_x \in \mathcal{I}$ lands in that interval: $\frac{P_{Z_x}(\mathcal{I}_j)}{P_{Z_x}(\mathcal{I})}$, and then sample uniformly from $\mathcal{I}_j$. If we have good control over the maximum of f_{Z_x} in $\mathcal{I}_j$, then we could use the ideas behind Theorem 2 to rejection sample from Z_x over $\mathcal{I}_j$, which would be much cheaper than rejection sampling from Z_x based on the uniform distribution over $[0, D]$. Furthermore, if we take Q to be uniform over $[0, d]$, then in Step 2 above we can transform $z_i \sim Q$ into $z'_i \sim Q_x$ by sampling $j \in [m]$ with probability $P_{Z_x}(\mathcal{I}_j)$ and setting $z'_i = z_i + (j-1)d$. Therefore, in Step 4 above, the short explanation the bounded leakage query needs to return to the simulator is simply j, which requires $\log m$ bits of bounded leakage.

The above almost works, except that over intervals $\mathcal{I}_j$ with extremely small mass $P_{Z_x}(\mathcal{I}_j)$ the guarantee on f_{Z_x} that we obtain are not enough for our end goal of bounding the ratio between f_{Z_x} and Q_x. To handle this, we modify the definition of Q_x so that

$$f_{Q_x}(z) = \frac{1}{2} \cdot \sum_{j=1}^{m} \left(\frac{P_{Z_x}(\mathcal{I}_j)}{P_{Z_x}(\mathcal{I})} \cdot \frac{1_{z \in \mathcal{I}_j}}{|\mathcal{I}_j|} \right) + \frac{1}{2} \cdot \frac{1_{z \in \mathcal{I}}}{|\mathcal{I}|}.$$

In other words, with probability $1/2$ we follow the procedure above, and with probability $1/2$ we sample uniformly from the whole $\mathcal{I} = [0, D]$. Intuitively, the second term will handle the intervals $\mathcal{I}_j$ with extremely small $P_{Z_x}(\mathcal{I}_j)$. Moreover, taking Q uniform over $[0, d]$, we only need $\log m$ additional bits of bounded leakage to turn $z_i \sim Q$ into $z'_i \sim Q_x$ in Step 2 above.

5.2 Splitting The Support

We introduce a generalization of Definition 3 to exploit different maximums in different sub-intervals of the support of the leakage Z. Later on, we will show that having a Lipschitz property on the pdfs of Z_x for $x \in \mathcal{X}$ is sufficient to appropriately bound all such maximums. This can simplify the computation of simulation parameters with respect to the more efficient simulator designed in this section.

For the sake of simplicity, we focus on the one-dimensional case where the leakage Z is supported on $\mathbb{R}$. These results are easily generalizable to Z supported on $\mathbb{R}^k$ for $k > 1$.

Definition 5. *Let Z be a random variable supported on $\mathbb{R}$. Let $\boldsymbol{\mathcal{I}} = (\mathcal{I}_1, \ldots, \mathcal{I}_m)$ be a collection of disjoint intervals in $\mathbb{R}$ and $\boldsymbol{M} = (M_{x,j})_{x \in \mathcal{X}, j \in [m]} \in \mathbb{R}^{|\mathcal{X}| \times m}$. Z is said to be $(\boldsymbol{\mathcal{I}}, \boldsymbol{M}, \delta)$-*noisy leakage *from X if*

- $f_{Z_x}(z) \leq M_{x,j}$ *for all* $z \in \mathcal{I}_j$, $j \in [m]$, *and* $x \in \mathcal{X}$,
- $\mathbb{E}_{x \sim P_X}\left[P_{Z_x}(\bigcup_{j \in [m]} \mathcal{I}_j)\right] \geq 1 - \delta$.

The following lemma is an important component of the correctness argument for our more efficient simulator in Algorithm 1.

Lemma 8. *Let $\mathcal{R}$ be a length D interval in $\mathbb{R}$, and let $\boldsymbol{\mathcal{I}} = (\mathcal{I}_1, \ldots, \mathcal{I}_m)$ be a partition of $\mathcal{R}$ into disjoint intervals each of length $d < D$. Suppose that Z is an $(\boldsymbol{\mathcal{I}}, \boldsymbol{M}, \delta)$-noisy leakage from X. For a given $x \in \mathcal{X}$ with $\delta_x = 1 - P_{Z_x}(\mathcal{R}) < 1$, consider a distribution Q_x with pdf f_{Q_x} satisfying*

$$f_{Q_x}(z) = \begin{cases} \frac{1}{2}\left(\frac{P_{Z_x}(\mathcal{I}_j)}{d(1-\delta_x)} + \frac{1}{D}\right), & \text{if } z \in \mathcal{I}_j, \\ 0, & \text{if } z \notin \mathcal{R}. \end{cases}$$

Then, for

$$t_x = \max_{j \in [m]} \log\left(2M_{x,j} \cdot \left[P_{Z_x}(\mathcal{I}_j) \cdot d^{-1} + D^{-1}\right]^{-1}\right)$$

it holds that $\mathsf{SD}_{t_x}(P_{Z_x};\, Q_x) \leq \delta_x$.

Proof. See Appendix B. □

We are now ready to prove the simulation theorem. For simplicity, we focus on one-dimensional leakage and on the case where $\mathcal{R}$ is an interval, but our argument generalizes beyond that which is presented in Appendix E. For a discussion on the complexity of oracle accesses, see Remark 1.

Function Leak $\left(x, (z_i)_{i\in[T]}\right)$
  for $i := 1$ **to** T **do**
    sample j_0 uniformly at random from $[m]$;
    $j_1 \leftarrow 0$;
    for $c := 1$ **to** $B = \log(T/\alpha)$ **do**
      sample y according to P_{Z_x};
      if $y \in [a, a+D]$ **then**
        Set $j_1 \in [m]$ to be the unique index such that $y \in \mathcal{I}_{j_1}$; `/* This loop attempts to sample from` $P_{Z_x|Z_x\in\mathcal{R}}$ `*/`
        break;
      end
    end
    if $j_1 = 0$ **then**
      return $(1,1)$; `/* If sampling from` $P_{Z_x|Z_x\in\mathcal{R}}$ `failed, abort and output default pair */`
    end
    sample b uniformly at random from $\{0,1\}$;
    $j \leftarrow j_b$;
    $z_i' \leftarrow z_i + (a + (j-1)\cdot d)$; `/* Shift` z_i `to` $\mathcal{I}_j$ `to get` $z_i' \sim Q_x$ `*/`
    if $i < T$ **then**
      with probability $\frac{f_{Z_x}(z_i')}{M_{x,j} f_{Q_x}(z_i')}$ **return** (i,j);
    end
    else
      return (T,j) ; `/* If first` $T-1$ `samples are rejected, the` T`-th sample is accepted */`
    end
  end
end
Function $\mathrm{Sim}^{\mathrm{Leak}(x,\cdot)}$
  Sample $z_1, \cdots, z_T$ independently and uniformy at random from $[0, d]$;
  $i, j \leftarrow$ Leak $\left(x, (z_i)_{i\in[T]}\right)$;
  $\tilde{z} \leftarrow z_i + (a + (j-1)\cdot d)$;
  return $\tilde{z}$;
end

Algorithm 1: The new rejection sampling simulator. The distribution Q_x is defined in the statement of Lemma 8.

Theorem 7. *Let $\mathcal{R} = [a, a + D] \subseteq \mathbb{R}$ for some $a, D \in \mathbb{R}$ and $\boldsymbol{\mathcal{I}} = (\mathcal{I}_1, \ldots, \mathcal{I}_m)$ be a partition of $\mathcal{R}$ into $m = D/d$ disjoint intervals of length d. Suppose that Z is $(\boldsymbol{\mathcal{I}}, \boldsymbol{M}, \delta)$-noisy leakage from X. Then, for any $\alpha > 0$, Z is $(2\alpha + 3\delta)$-simulatable via Algorithm 1 from*

$$\ell = \lceil t + \log \ln(1/\alpha) + \log(D/d) \rceil$$

bits of bounded leakage from X, where

$$t = \max_{x \in \mathcal{X}, j \in [m]} \log \left(2M_{x,j} \cdot \left[P_{Z_x}(\mathcal{I}_j) \cdot d^{-1} + D^{-1} \right]^{-1} \right).$$

Furthermore, the simulator from Algorithm 1 runs in time $O(2^t \log(1/\alpha) \cdot (t + \log(1/\alpha)))$ assuming oracle access to the pdfs of the uniform distribution over $[a, a + D]$ and P_{Z_x} for every $x \in \mathcal{X}$, and oracle access to i.i.d. samples from $[0, d]$ and P_{Z_x}.

Proof. Fix $\alpha > 0$. We run the simulator from Algorithm 1 with $T = 2^t \ln(1/\alpha)$. First, we claim that Lines 3–17 in Algorithm 1 transform a sample z_i into a correct sample z_i' of Q_x, for all $i = 1, \ldots, T$, except with probability $2\delta + \alpha$ over $x \sim P_X$ and the randomness of the algorithm. Observe that if after Lines 4–11 we have $j_1 \neq 0$, then $\Pr[j_1 = j^\star] = \frac{P_{Z_x}(\mathcal{I}_{j^\star})}{1 - \delta_x}$ for any $j^\star \in [m]$. Suppose that $j_1 \neq 0$. With probability $1/2$ we have $j = j_1$. In this case, since Z_i is uniformly random over $[0, d]$, after fixing j the random variable $Z_i' = Z_i + (a + (j-1)d)$ is uniformly distributed over $\mathcal{I}_j$. Therefore, Z_i' conditioned only on $b = 1$ on Line 5 assigns density $\frac{P_{Z_x}(\mathcal{I}_j)}{d(1-\delta_x)}$ to all $z \in \mathcal{I}_j$. On the other hand, with probability $1/2$ we set $j = j_0$, and so in this case j is uniformly random over $[m]$. Since z_i is uniformly random over $[0, d]$, it follows that $z_i' = z_i + (a + (j-1)d)$ is uniformly random over $[0, D]$.

Putting together the observations above shows that z_i' is distributed according to Q_x provided that Lines 4–11 produce $j_1 \neq 0$. We proceed to upper bound the probability that $j_1 = 0$ on Line 12 for at least one $i \in [T]$. Call x *good* if $\delta_x \leq 1/2$, and *bad* otherwise. Since $\mathbb{E}_{x \sim P_X}[\delta_x] \leq \delta$, an averaging argument gives that $\Pr_{x \sim P_X}[x \text{ is bad}] \leq 2\delta$. In particular, for x good we have

$$\Pr_{y \sim P_{Z_x}}[y \notin [a, a + D]] = \delta_x \leq 1/2.$$

Suppose that x is good and fix an arbitrary $i \in [T]$. The probability that $j_1 = 0$ on Line 12 in the i-th iteration of the For loop on Line 2 is

$$\delta_x^B \leq 2^{-B} = \alpha/T$$

by our choice of $B = \log(T/\alpha)$. Then, a union bound over the T iterations gives that we have $j_1 = 0$ on Line 12 for at least one iteration $i \in [T]$ with probability at most $T \cdot \alpha/T = \alpha$.

Suppose that $j_1 \neq 0$ holds for all $i \in [T]$. Then, the bounded leakage query $\mathrm{Leak}(x, (z_i)_{i \in [T]})$ in Algorithm 1 performs rejection sampling from P_{Z_x} based

on the z_i''s, which the paragraphs above showed are i.i.d. samples from Q_x. Combining Lemma 8 and Theorem 1 with our choice of $T = 2^t \ln(1/\alpha)$ gives that this rejection sampling procedure with $L = 2^t \ln(1/\alpha)$ samples produces an output distribution $P_{\tilde{Z}_x}$ that is $\epsilon_x = \max(\delta_x, (1-(1-\delta_x)2^{-t})^T)$ close to P_{Z_x} in statistical distance. It follows that

$$\begin{aligned}
&\mathsf{SD}(P_{XZ}; P_{X\tilde{Z}}) = \mathbb{E}_{x\sim P_X}[\mathsf{SD}(P_{Z_x}; P_{\tilde{Z}_x})] \\
&\leq \Pr_{x\sim P_X}[x \text{ is bad}] + \sum_{x \text{ is good}} P_X(x) \cdot \mathsf{SD}(P_{Z_x}; P_{\tilde{Z}_x}) \\
&\leq \Pr_{x\sim P_X}[x \text{ is bad}] \\
&+ \sum_{x \text{ is good}} P_X(x) \cdot (\Pr[\exists i \in [T] \text{ s.t. } j_1 = 0 | X = x] + \epsilon_x) \\
&\leq 2\delta + \alpha + \sum_{x \text{ is good}} P_X(x) \cdot \epsilon_x \\
&\leq 2\delta + \alpha + \mathbb{E}_{x\sim P_X}[\epsilon_x] \leq 3\delta + 2\alpha.
\end{aligned}$$

The second inequality follows from the fact that for good x we have either $j_1 = 0$ for some iteration $i \in [T]$, or, by the paragraph above, $P_{\tilde{Z}_x}$ and P_{Z_x} are ϵ_x-close in statistical distance. The third inequality uses the fact that $\Pr_{x\sim P_X}[x \text{ is bad}] \leq 2\delta$ and that $\Pr[j_1 = 0 \text{ for some } i \in [T] | X = x] \leq \alpha$ for all good x. The last inequality uses the fact from Theorem 1 that $\mathbb{E}_{x\sim P_X}[\epsilon_x] \leq \alpha + \delta$ given $\mathbb{E}_x[\delta_x] \leq \delta$.

It remains to account for the total amount of bounded leakage output by the Leak procedure and the overall running time of the simulator. Revealing the accepted index in the rejection sampling procedure requires $\log(T \ln(1/\alpha)) = t + \log\ln(1/\alpha)$ bits of bounded leakage. Additionally revealing j requires $\log m = \log(D/d)$ bits of bounded leakage. Therefore, the bounded leakage query leaks a total of $\ell = \lceil t + \log\ln(1/\alpha) + \log(D/d) \rceil$ bits.

Regarding the running time of the simulator, first recall that we assume oracle (i.e., $O(1)$) access to i.i.d. samples from the uniform distribution over $[0, d]$ and from P_{Z_x}, and also oracle access to the pdfs of the uniform distribution over $[a, a+D]$ and of P_{Z_x} for every $x \in \mathcal{X}$. Note that the operations in all lines of Algorithm 1 take time $O(1)$. Therefore, the running time of Algorithm 1 is $O(T \cdot B) = O(2^t \log(1/\alpha) \cdot (t + \log(1/\alpha)))$. □

5.3 Lipschitz-Noisy Leakage

Theorem 7 tells us that we can get a bounded leakage vs. simulation efficiency tradeoff as long as we control the maximum of f_{Z_x} in every sub-interval $\mathcal{I}_j$.

For this, we assume the γ-Lipschitz property on the pdfs f_{Z_x}. This motivates to introduce the $(\mathcal{R}, \gamma, \delta)$-Lipschitz-noisy leakage:

Definition 6 (Lipschitz-noisy leakage). *Let X and Z be random variables with Z supported on $\mathbb{R}^k$. Let $\mathcal{R}$ be a region over $\mathbb{R}^k$. We say that Z is $(\mathcal{R}, \gamma, \delta)$-*

Lipschitz-noisy leakage *from X if for each $x \in \mathcal{X}$ the pdf f_{Z_x} is γ-Lipschitz* [10] *and* $\mathbb{E}_{x\sim P_X}[P_{Z_x}(\mathcal{R})] \geq 1-\delta$.

First, we consider the one-dimesnional leakage case. The extended framework in multi-dimensions is presented in Appendix E. We will soon show that the family of $(\mathcal{R},\gamma,\delta)$-Lipschitz-noisy leakages from X is a subfamily of $(\mathcal{R}, M=\sqrt{2\gamma},\delta)$-noisy leakages from X. It will also allow us to bound maximums per interval (this somewhat mirrors the Lipschitz's functions behavior under Riemann's integral).

Furthermore, computing the γ parameter in the definition of $(\mathcal{R},\gamma,\delta)$-Lipschitz-noisy leakage is simple if it is easy to compute and bound the derivative of f_{Z_x}. For example, this is the case in the "additive noise" case discussed throughout this work where $Z = \mathsf{d}(X) + \mathcal{N}$ with $\mathcal{N}$ independent noise with a well-behaved pdf $f_{\mathcal{N}}$. Typically, $\gamma = (\sqrt{2\pi e\sigma^2})^{-1}$ when $\mathcal{N} = \mathcal{N}(0,\sigma^2)$.

The following lemma formally captures the connection between the γ-Lipschitz property and the maximum over an interval.

Lemma 9. *Let $f : \mathbb{R} \to \mathbb{R}_{\geq 0}$ be γ-Lipschitz. Fix an interval $\mathcal{I} \subseteq \mathbb{R}$ of length d and let $\beta = \int_{\mathcal{I}} f(x)\,dx$. Then, $\max_{x\in\mathcal{I}} f(x) \leq \max(\sqrt{2\beta\gamma}, 2\beta/d)$.*

Proof. A direct corollary of Lemma 10 in Appendix E with dimension $k = 1$. □

We can now combine Lemma 9 with Theorem 7 to get a simulation theorem for Lipschitz-noisy leakage.

Theorem 8. *Let $\mathcal{R} = [a, a+D]$ for some $a, D \in \mathbb{R}$. Let $\boldsymbol{\mathcal{I}} = (\mathcal{I}_1, \ldots, \mathcal{I}_m)$ be a partition of $\mathcal{R}$ into $m = D/d$ intervals of length d. Suppose that Z is $(\mathcal{R},\gamma,\delta)$-Lipschitz-noisy leakage from X. Then, Z is also $(\boldsymbol{\mathcal{I}}, \boldsymbol{M}, \delta)$-noisy leakage from X with*

$$M_{x,j} = \max\left(\sqrt{2P_{Z_x}(\mathcal{I}_j)\gamma}, 2P_{Z_x}(\mathcal{I}_j)/d\right).$$

In particular, Z is ε-simulatable from

$$\ell = \max\Bigg(\left\lceil 2 + \log(D/d) + \log\ln\left(\frac{2}{\varepsilon-3\delta}\right)\right\rceil, \left\lceil \log D + \frac{1}{2}\log(2\gamma) + \frac{1}{2}\log\left(D/d\right) + \log\ln\left(\frac{2}{\varepsilon-3\delta}\right) + 1\right\rceil\Bigg)$$

bits of bounded leakage from X.

Furthermore, the simulation runs in time $O(1 + \sqrt{Dd\gamma}\cdot\log(2/(\varepsilon-3\delta))\cdot\left[\log(Dd\gamma) + \log(2/(\varepsilon-3\delta))\right])$ assuming oracle access to the pdfs of the uniform distribution over $[a, a+D]$ and P_{Z_x} for every $x \in \mathcal{X}$, and oracle access to i.i.d. samples from $[0,d]$ and P_{Z_x}.

Proof. See Appendix D. □

[10] A function $f : \mathbb{R}^k \to \mathbb{R}$ is *γ-Lipschitz* if $|f(\mathbf{x_1}) - f(\mathbf{x_0})| \leq \gamma\|\mathbf{x_1} - \mathbf{x_0}\|$ for all $\mathbf{x_0}, \mathbf{x_1} \in \mathbb{R}^k$. Note, we can consider the standard ℓ^2-norm or ℓ^∞-norm here.

Simulation efficiency vs. bounded leakage tradeoffs. We now discuss the tradeoff between simulation efficiency and bounded leakage that can be achieved via Theorem 8 compared to our original approach in Theorem 2. For simplicity, we focus on the case of $(\mathcal{R}, \gamma, \delta)$-Lipschitz-noisy leakages Z from X, as the main improvements and underlying ideas are already apparent in this setting.

We begin by working out the application of Theorem 2 to simulating Z. If Z is $(\mathcal{R}, \gamma, \delta)$-Lipschitz-noisy leakage from X, then $f_{Z_x}(z) \leq \sqrt{2\gamma}$ for all $z \in \mathbb{R}$ and $x \in \mathcal{X}$ (for example, this is implicit in the first case of the proof of Lemma 9). Suppose that for $\mathcal{R} = [a, a + D] \subseteq \mathbb{R}$ we have $\mathbb{E}_{x \sim P_X}[P_{Z_x}(\mathcal{R})] \geq 1 - \delta$.

Then, it follows that Z is $(\mathcal{R}, M = \sqrt{2\gamma}, \delta)$-noisy leakage from X, and so Theorem 2 guarantees that Z is ε-simulatable from

$$\ell_1 = \left\lceil \log \left\lceil \frac{\ln \frac{1}{\varepsilon}}{-\ln(1 - \frac{1-\delta}{D\sqrt{2\gamma}})} \right\rceil \right\rceil \approx \log \ln \frac{1}{\varepsilon} + \log D + \frac{1}{2} \log(2\gamma) + \delta$$

bits of bounded leakage from X.

Furthermore, this simulation runs in time $\Theta(D\sqrt{\gamma} \ln(1/\varepsilon))$, assuming oracle accesses.

On the other hand, applying Theorem 8 to Z, and assuming that the product $Dd\gamma$ is not extremely small (this ensures the second term dominates the first term in the expression of ℓ in Theorem 8), we conclude that Z is ε-simulatable from

$$\ell_2 = \left\lceil \log D + \frac{1}{2} \log(2\gamma) + \frac{1}{2} \log (D/d) + \log \ln \left(\frac{2}{\varepsilon - 3\delta} \right) + 1 \right\rceil$$

bits of bounded leakage from X.

Furthermore, this simulation runs in time $O\big(1 + \sqrt{Dd\gamma} \cdot \log(2/(\varepsilon - 3\delta)) \cdot [\log(Dd\gamma) + \log(2/(\varepsilon - 3\delta))]\big)$, ignoring oracle accesses.

We can observe, that the amount of extra bits of bounded leakage when going from Theorem 8 to Theorem 2 is $\Delta = \ell_2 - \ell_1 \approx \frac{1}{2} \log(D/d)$, while the simulation time drops from $O(D\sqrt{\gamma})$ to $O(\sqrt{Dd\gamma})$ (omitting $\log(1/\varepsilon)$ factors in both).

As one illustration of this tradeoff, suppose that $D = 2^{\Theta(\lambda)}$ and $D\gamma = 2^{\Theta(\lambda)}$ and that we target a simulation error of $\varepsilon = 2^{-\lambda}$. In particular, simulation via Theorem 2 is inefficient in this case, requiring simulation time $2^{\Theta(\lambda)}$. If we set $d = \frac{1}{D\gamma} = 2^{-\Omega(\lambda)}$, then we obtain simulation running time

$$O(1 + \sqrt{Dd\gamma} \cdot \log(2/(\varepsilon - 3\delta)) \cdot [\log(Dd\gamma) + \log(2/(\varepsilon - 3\delta))]) = O(\lambda^2)$$

at the cost of an additional

$$\frac{1}{2} \log(D/d) = \log D + \frac{1}{2} \log \gamma = O(\lambda)$$

bits of bounded leakage from X.

6 Overview of Applications

We explore applications of simulation theorems for noisy leakage from bounded leakage to obtaining leakage-resilient cryptographic primitives in the *computational* setting. Following the literature [33], we distinguish between applications in which either the memory or the computation leaks information. For space reasons, we only provide a brief overview here and refer the reader to the full version [5] for formal statements and proofs.

6.1 Leakage from Computation

A prominent line of research in the setting of computation leakage aims at constructing specific primitives which exploit the existence of a leak-free memory: the values that are stored in the memory cannot leak, but whenever some of these values are moved from the memory to the CPU in order to perform some computations, they become subject to leakage. This assumption was introduced for the first time by Micali and Reyzin [40] and became known as the axiom that only computation leaks information.

For concreteness, let us consider the case of leakage-resilient stream ciphers. A stream cipher is a function $\mathsf{sc} : \{0,1\}^n \to \{0,1\}^n \times \{0,1\}^m$ that for every secret state $X_0 \in \{0,1\}^n$ defines a sequence of $q+1$ outputs $Y_1, \ldots, Y_{q+1} \in \{0,1\}^m$, for $q \in \mathbb{N}$, that are determined recursively as $(X_{i+1}, Y_{i+1}) = \mathsf{sc}(X_i)$. The standard security notion of a stream cipher requires that, for random $X_0 = U_n$, the outputs $Y_1, \ldots, Y_{q+1}$ are pseudorandom. Leakage resilience, on the other hand, requires that Y_{q+1} is pseudorandom even given $Y_1, \ldots, Y_q$ along with the leakages $Z_1, \ldots, Z_q$, where $Z_i = f_i(X_i^+)$ and X_i^+ is the part of the state that is needed in the i-th round of the computation.

The first construction of a bounded leakage-resilient stream cipher was proposed by Dziembowski and Pietrzak [23] and later simplified by Pietrzak [44]. These initial results were improved in subsequent works [53, 26, 52]. In the full version [5], we show that any stream cipher secure against bounded non-adaptive leakage is also secure against noisy non-adaptive leakage via any family $\mathcal{F}$ that is simulatable from bounded leakage with efficient simulation and small error.

6.2 Leakage from Memory

In the memory leakage model, the attacker obtains some leakage from the memory of a cryptosystem, which usually contains the secret key $X \in \mathcal{X}$. Most of the results consider ℓ-bounded leakage, which basically means that the leakage consists of at most ℓ bits for some parameter $\ell \in \mathbb{N}$ (shorter than the length of X). In the full version [5], we study memory leakage resilience for arbitrary cryptographic primitives and protocols with either game-based or simulation-based security. In particular, we show that any cryptosystem secure in the presence of bounded leakage (as defined above) is also secure in the presence of noisy leakage, so long as we can simulate the noisy leakage with bounded leakage efficiently and with negligible simulation error. The above statement holds for a

large family of leakage-resilient primitives, including public-key encryption [1, 41, 17, 29], digital signatures [34, 25, 17, 27, 12, 42, 15, 24], and identity-based encryption [37], as well as leakage-tolerant interactive protocols for different ideal functionalities, including coin tossing [11], zero-knowledge [28, 8], secure message transmission, message authentication, commitments, and oblivious transfer [8].

Conclusion

In this work, we modified the simulator from [43] to provide easier-to-determine and better parameters. A first approach, based on bounding the volume of a parallelotope containing the leakage with overwhelming probability, already yields nice improvements despite very simple. A second approach, based on sampling the *likelihest conditional*, provides even better parameters which can be expressed in terms of maximal leakage, but at the cost of a more complex simulation. Finally, a third approach trades extra required bits of bounded leakage for a better simulation efficiency, by splitting the support of the simulation into multiple sub-intervals. These results have direct applications to leakage-resilient cryptographic primitives as mentioned in Section 6. In turn, they narrow the gap for the applicability of reductions from noisy leakages to bounded leakages.

Acknowledgments

François-Xavier Standaert is a research director from the Belgian fund for Scientific Research (FNRS-F.R.S.). Daniele Venturi is a member of the Gruppo Nazionale Calcolo Scientifico Istituto Nazionale di Alta Matematica (GNCS-INdAM). His research was partially supported by project BEAT—funded by Sapienza University of Rome—and by the AI-Sec Lab initiative between Sapienza University of Rome and the CYBER 4.0 Competence Center. João Ribeiro was supported by national funds through FCT – Fundação para a Ciência e a Tecnologia, I.P., and, when eligible, co-funded by EU funds under project/support UID/50008/2025 – Instituto de Telecomunicações, with DOI 10.54499/UID/50008/2025. Ananta Mukherjee was supported by National Quantum Scholarship Scheme, Singapore. This work has been funded in parts by the ERC Advanced Grant BRIDGE (101096871). Views and opinions expressed are those of the authors only and do not necessarily reflect those of the EU or the ERC. Neither the EU nor the granting authority can be held responsible for them.

References

1. Akavia, A., Goldwasser, S., Vaikuntanathan, V.: Simultaneous hardcore bits and cryptography against memory attacks. In: Reingold, O. (ed.) TCC 2009. LNCS, vol. 5444, pp. 474–495. Springer, Berlin, Heidelberg (Mar 2009). https://doi.org/10.1007/978-3-642-00457-5_28

2. Ananth, P., Goyal, V., Pandey, O.: Interactive proofs under continual memory leakage. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part II. LNCS, vol. 8617, pp. 164–182. Springer, Berlin, Heidelberg (Aug 2014). https://doi.org/10.1007/978-3-662-44381-1_10
3. Archambeau, C., Peeters, E., Standaert, F., Quisquater, J.: Template attacks in principal subspaces. In: CHES. Lecture Notes in Computer Science, vol. 4249, pp. 1–14. Springer (2006)
4. Béguinot, J., Cheng, W., Guilley, S., Rioul, O.: Formal security proofs via Doeblin coefficients: Optimal side-channel factorization from noisy leakage to random probing. In: Reyzin, L., Stebila, D. (eds.) Advances in Cryptology – CRYPTO 2024. pp. 389–426. Springer Nature Switzerland, Cham (2024)
5. Béguinot, J., Mukherjee, A., Obresmki, M., Ribeiro, J., Roy, L., Standaert, F., Venturi, D.: Simulating noisy leakage with bounded leakage: Simpler, better, faster. IACR Cryptol. ePrint Arch. **2026**, 357 (2026), https://eprint.iacr.org/2026/357
6. Belaïd, S., Cassiers, G., Mutschler, C., Rivain, M., Roche, T., Standaert, F., Taleb, A.R.: Sok: A methodology to achieve provable side-channel security in real-world implementations. IACR Commun. Cryptol. **2**(1), 4 (2025)
7. Benhamouda, F., Degwekar, A., Ishai, Y., Rabin, T.: On the local leakage resilience of linear secret sharing schemes. Journal of Cryptology **34**(2), 10 (Apr 2021). https://doi.org/10.1007/s00145-021-09375-2
8. Bitansky, N., Canetti, R., Halevi, S.: Leakage-tolerant interactive protocols. In: Cramer, R. (ed.) TCC 2012. LNCS, vol. 7194, pp. 266–284. Springer, Berlin, Heidelberg (Mar 2012). https://doi.org/10.1007/978-3-642-28914-9_15
9. Boyle, E., Garg, S., Jain, A., Kalai, Y.T., Sahai, A.: Secure computation against adaptive auxiliary information. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part I. LNCS, vol. 8042, pp. 316–334. Springer, Berlin, Heidelberg (Aug 2013). https://doi.org/10.1007/978-3-642-40041-4_18
10. Boyle, E., Goldwasser, S., Jain, A., Kalai, Y.T.: Multiparty computation secure against continual memory leakage. In: Karloff, H.J., Pitassi, T. (eds.) 44th ACM STOC. pp. 1235–1254. ACM Press (May 2012). https://doi.org/10.1145/2213977.2214087
11. Boyle, E., Goldwasser, S., Kalai, Y.T.: Leakage-resilient coin tossing. Distributed Comput. **27**(3), 147–164 (2014). https://doi.org/10.1007/S00446-013-0206-Z
12. Boyle, E., Segev, G., Wichs, D.: Fully leakage-resilient signatures. Journal of Cryptology **26**(3), 513–558 (Jul 2013). https://doi.org/10.1007/s00145-012-9136-3
13. Brakerski, Z., Kalai, Y.T., Katz, J., Vaikuntanathan, V.: Overcoming the hole in the bucket: Public-key cryptography resilient to continual memory leakage. In: 51st FOCS. pp. 501–510. IEEE Computer Society Press (Oct 2010). https://doi.org/10.1109/FOCS.2010.55
14. Brian, G., Faonio, A., Obremski, M., Ribeiro, J., Simkin, M., Skórski, M., Venturi, D.: The mother of all leakages: How to simulate noisy leakages via bounded leakage (almost) for free. IEEE Transactions on Information Theory **68**(12), 8197–8227 (2022). https://doi.org/10.1109/TIT.2022.3193848, preliminary version in Eurocrypt 2021.
15. Dagdelen, Ö., Venturi, D.: A second look at Fischlin's transformation. In: Pointcheval, D., Vergnaud, D. (eds.) AFRICACRYPT 14. LNCS, vol. 8469, pp. 356–376. Springer, Cham (May 2014). https://doi.org/10.1007/978-3-319-06734-6_22
16. Dodis, Y., Haralambiev, K., López-Alt, A., Wichs, D.: Cryptography against continuous memory attacks. In: 51st FOCS. pp. 511–520. IEEE Computer Society Press (Oct 2010). https://doi.org/10.1109/FOCS.2010.56

17. Dodis, Y., Haralambiev, K., López-Alt, A., Wichs, D.: Efficient public-key cryptography in the presence of key leakage. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 613–631. Springer, Berlin, Heidelberg (Dec 2010). https://doi.org/10.1007/978-3-642-17373-8_35
18. Dodis, Y., Lewko, A.B., Waters, B., Wichs, D.: Storing secrets on continually leaky devices. In: Ostrovsky, R. (ed.) 52nd FOCS. pp. 688–697. IEEE Computer Society Press (Oct 2011). https://doi.org/10.1109/FOCS.2011.35
19. Dodis, Y., Yu, Y.: Overcoming weak expectations. In: Sahai, A. (ed.) TCC 2013. LNCS, vol. 7785, pp. 1–22. Springer, Berlin, Heidelberg (Mar 2013). https://doi.org/10.1007/978-3-642-36594-2_1
20. Duc, A., Dziembowski, S., Faust, S.: Unifying leakage models: From probing attacks to noisy leakage. J. Cryptol. **32**(1), 151–177 (2019)
21. Duc, A., Faust, S., Standaert, F.X.: Making masking security proofs concrete. In: Oswald, E., Fischlin, M. (eds.) Advances in Cryptology – EUROCRYPT 2015. pp. 401–429. Springer Berlin Heidelberg, Berlin, Heidelberg (2015)
22. Dziembowski, S., Faust, S., Skorski, M.: Noisy leakage revisited. In: Oswald, E., Fischlin, M. (eds.) Advances in Cryptology - EUROCRYPT 2015. pp. 159–188. Springer Berlin Heidelberg, Berlin, Heidelberg (2015)
23. Dziembowski, S., Pietrzak, K.: Leakage-resilient cryptography. In: 49th FOCS. pp. 293–302. IEEE Computer Society Press (Oct 2008). https://doi.org/10.1109/FOCS.2008.56
24. Faonio, A., Nielsen, J.B., Venturi, D.: Mind your coins: Fully leakage-resilient signatures with graceful degradation. In: Halldórsson, M.M., Iwama, K., Kobayashi, N., Speckmann, B. (eds.) ICALP 2015, Part I. LNCS, vol. 9134, pp. 456–468. Springer, Berlin, Heidelberg (Jul 2015). https://doi.org/10.1007/978-3-662-47672-7_37
25. Faust, S., Kiltz, E., Pietrzak, K., Rothblum, G.N.: Leakage-resilient signatures. In: Micciancio, D. (ed.) TCC 2010. LNCS, vol. 5978, pp. 343–360. Springer, Berlin, Heidelberg (Feb 2010). https://doi.org/10.1007/978-3-642-11799-2_21
26. Faust, S., Pietrzak, K., Schipper, J.: Practical leakage-resilient symmetric cryptography. In: Prouff, E., Schaumont, P. (eds.) CHES 2012. LNCS, vol. 7428, pp. 213–232. Springer, Berlin, Heidelberg (Sep 2012). https://doi.org/10.1007/978-3-642-33027-8_13
27. Galindo, D., Vivek, S.: A leakage-resilient pairing-based variant of the Schnorr signature scheme. In: Stam, M. (ed.) 14th IMA International Conference on Cryptography and Coding. LNCS, vol. 8308, pp. 173–192. Springer, Berlin, Heidelberg (Dec 2013). https://doi.org/10.1007/978-3-642-45239-0_11
28. Garg, S., Jain, A., Sahai, A.: Leakage-resilient zero knowledge. In: Rogaway, P. (ed.) CRYPTO 2011. LNCS, vol. 6841, pp. 297–315. Springer, Berlin, Heidelberg (Aug 2011). https://doi.org/10.1007/978-3-642-22792-9_17
29. Hazay, C., López-Alt, A., Wee, H., Wichs, D.: Leakage-resilient cryptography from minimal assumptions. Journal of Cryptology **29**(3), 514–551 (Jul 2016). https://doi.org/10.1007/s00145-015-9200-x
30. Ishai, Y., Sahai, A., Wagner, D.: Private circuits: Securing hardware against probing attacks. In: Boneh, D. (ed.) Advances in Cryptology - CRYPTO 2003. pp. 463–481. Springer Berlin Heidelberg, Berlin, Heidelberg (2003)
31. Issa, I., Wagner, A.B., Kamath, S.: An operational approach to information leakage. IEEE Trans. Inf. Theory **66**(3), 1625–1657 (2020). https://doi.org/10.1109/TIT.2019.2962804, https://doi.org/10.1109/TIT.2019.2962804
32. Kac, M., Murdock, W.W., Szegő, G.: On the eigen-values of certain hermitian forms. Indiana University Mathematics Journal **2**, 767–800 (1953), https://api.semanticscholar.org/CorpusID:124672462

33. Kalai, Y.T., Reyzin, L.: A survey of leakage-resilient cryptography. In: Providing Sound Foundations for Cryptography: On the Work of Shafi Goldwasser and Silvio Micali, pp. 727–794. ACM (2019)
34. Katz, J., Vaikuntanathan, V.: Signature schemes with bounded leakage resilience. In: Matsui, M. (ed.) ASIACRYPT 2009. LNCS, vol. 5912, pp. 703–720. Springer, Berlin, Heidelberg (Dec 2009). https://doi.org/10.1007/978-3-642-10366-7_41
35. Kellerer, H., Mansini, R., Pferschy, U., Speranza, M.G.: An efficient fully polynomial approximation scheme for the subset-sum problem. J. Comput. Syst. Sci. **66**(2), 349–370 (2003). https://doi.org/10.1016/S0022-0000(03)00006-0, https://doi.org/10.1016/S0022-0000(03)00006-0
36. Lewko, A.B., Lewko, M., Waters, B.: How to leak on key updates. In: Fortnow, L., Vadhan, S.P. (eds.) 43rd ACM STOC. pp. 725–734. ACM Press (Jun 2011). https://doi.org/10.1145/1993636.1993732
37. Lewko, A.B., Rouselakis, Y., Waters, B.: Achieving leakage resilience through dual system encryption. In: Ishai, Y. (ed.) TCC 2011. LNCS, vol. 6597, pp. 70–88. Springer, Berlin, Heidelberg (Mar 2011). https://doi.org/10.1007/978-3-642-19571-6_6
38. Mangard, S.: Hardware countermeasures against DPA ? A statistical analysis of their effectiveness. In: CT-RSA. Lecture Notes in Computer Science, vol. 2964, pp. 222–235. Springer (2004)
39. Mangard, S., Oswald, E., Popp, T.: Power analysis attacks - revealing the secrets of smart cards. Springer (2007)
40. Micali, S., Reyzin, L.: Physically observable cryptography (extended abstract). In: Naor, M. (ed.) TCC 2004. LNCS, vol. 2951, pp. 278–296. Springer, Berlin, Heidelberg (Feb 2004). https://doi.org/10.1007/978-3-540-24638-1_16
41. Naor, M., Segev, G.: Public-key cryptosystems resilient to key leakage. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 18–35. Springer, Berlin, Heidelberg (Aug 2009). https://doi.org/10.1007/978-3-642-03356-8_2
42. Nielsen, J.B., Venturi, D., Zottarel, A.: Leakage-resilient signatures with graceful degradation. In: Krawczyk, H. (ed.) PKC 2014. LNCS, vol. 8383, pp. 362–379. Springer, Berlin, Heidelberg (Mar 2014). https://doi.org/10.1007/978-3-642-54631-0_21
43. Obremski, M., Ribeiro, J., Roy, L., Standaert, F.X., Venturi, D.: Improved reductions from noisy to bounded and probing leakages via hockey-stick divergences. In: Reyzin, L., Stebila, D. (eds.) Advances in Cryptology – CRYPTO 2024. pp. 461–491. Springer Nature Switzerland, Cham (2024)
44. Pietrzak, K.: A leakage-resilient mode of operation. In: Joux, A. (ed.) EUROCRYPT 2009. LNCS, vol. 5479, pp. 462–482. Springer, Berlin, Heidelberg (Apr 2009). https://doi.org/10.1007/978-3-642-01001-9_27
45. Prest, T., Goudarzi, D., Martinelli, A., Passelègue, A.: Unifying leakage models on a Rényi day. In: Boldyreva, A., Micciancio, D. (eds.) Advances in Cryptology – CRYPTO 2019. pp. 683–712. Springer International Publishing, Cham (2019)
46. Prouff, E., Rivain, M.: Masking against side-channel attacks: A formal security proof. In: Johansson, T., Nguyen, P.Q. (eds.) Advances in Cryptology – EUROCRYPT 2013. pp. 142–159. Springer Berlin Heidelberg, Berlin, Heidelberg (2013)
47. Schindler, W., Lemke, K., Paar, C.: A stochastic model for differential side channel cryptanalysis. In: CHES. Lecture Notes in Computer Science, vol. 3659, pp. 30–46. Springer (2005)
48. Shamos, M.I.: Computational geometry. (1978), https://api.semanticscholar.org/CorpusID:265963533

49. Sibson, R.: Information radius. Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete **14**, 149–160 (1969), https://api.semanticscholar.org/CorpusID:198179065
50. Toussaint, G.T.: Solving geometric problems with the rotating calipers (1983), https://api.semanticscholar.org/CorpusID:117404139
51. Udvarhelyi, B., van Wassenhove, A., Bronchain, O., Standaert, F.: On the security of off-the-shelf microcontrollers: Hardware is not enough. In: CARDIS. Lecture Notes in Computer Science, vol. 12609, pp. 103–118. Springer (2020)
52. Yu, Y., Standaert, F.X.: Practical leakage-resilient pseudorandom objects with minimum public randomness. In: Dawson, E. (ed.) CT-RSA 2013. LNCS, vol. 7779, pp. 223–238. Springer, Berlin, Heidelberg (Feb / Mar 2013). https://doi.org/10.1007/978-3-642-36095-4_15
53. Yu, Y., Standaert, F.X., Pereira, O., Yung, M.: Practical leakage-resilient pseudorandom generators. In: Al-Shaer, E., Keromytis, A.D., Shmatikov, V. (eds.) ACM CCS 2010. pp. 141–151. ACM Press (Oct 2010). https://doi.org/10.1145/1866307.1866324

Supplemental Material

A Proof of Lemma 1

Our proof follows that of [43, Theorem 4] closely. Let R be random variable produced by the rejection sampling algorithm. The probability distribution of the index of the returned sample follows a truncated geometric distribution. For one pass, the probability that a sample is accepted is

$$\begin{aligned} p &= \int_z f_Q(z) \min(1, \frac{f_P(z)}{2^t f_Q(z)}) dz \\ &= \int_z \min(f_Q(z), 2^{-t} f_P(z)) dz \\ &= 2^{-t} \int_z \min(2^t f_Q(z), f_P(z)) dz \\ &= 2^{-t}(1 - \mathsf{SD}_t(P\|Q)) \\ &= 2^{-t}(1-\delta). \end{aligned}$$

The probability that the sample is rejected is

$$r \triangleq 1 - p = 1 - (1-\delta)2^{-t}.$$

We can check that the probability of the output distribution is

$$\begin{aligned} f_R(z) &= \sum_{i=0}^{T-2} r^i f_Q(z) \min(1, \frac{f_P(z)}{2^t f_Q(z)}) + r^{T-1} f_Q(z) \\ &= \sum_{i=0}^{T-2} r^i \min(f_Q(z), 2^{-t} f_P(z)) + r^{T-1} f_Q(z) \end{aligned}$$

$$
\begin{aligned}
&\geq \sum_{i=0}^{T-1} r^i \min(f_Q(z), 2^{-t} f_P(z)) \\
&= \frac{1-r^T}{1-r} \min(f_Q(z), 2^{-t} f_P(z)).
\end{aligned}
$$

Now,

$$
\begin{aligned}
\mathsf{SD}(P;R) &= \int_z \max(0, f_P(z) - f_R(z))dz \\
&\leq \int_z \max\left(0, f_P(z) - \frac{1-r^T}{1-r} \min(f_Q(z), 2^{-t} f_P(z))\right) dz \\
&\leq \int_z \max\left(0, f_P(z) - \min\left(\frac{1-r^T}{1-r}, 2^t\right) \min(f_Q(z), 2^{-t} f_P(z))\right) dz \\
&= \int_z \max\left(0, f_P(z) - \min\left(\frac{1-r^T}{1-r} 2^{-t}, 1\right) \min(2^t f_Q(z), f_P(z))\right) dz \\
&= \int_z \max\left(0, f_P(z) - \min\left(\frac{1-r^T}{1-r} 2^{-t}, 1\right) (f_P(z) + \min(2^t f_Q(z) - f_P(z), 0))\right) dz \\
&= \int_z \max(0, f_P(z)(1 - \min(\frac{1-r^T}{1-r} 2^{-t}, 1)) - \min(\frac{1-r^T}{1-r} 2^{-t}, 1) \min(2^t f_Q(z) - f_P(z), 0))dz \\
&= \int_z \max(0, f_P(z) \max(1 - \frac{1-r^T}{1-r} 2^{-t}, 0) - \min(\frac{1-r^T}{1-r} 2^{-t}, 1) \min(2^t f_Q(z) - f_P(z), 0))dz \\
&\leq \max(1 - \frac{1-r^T}{1-r} 2^{-t}, 0) - \int_z \max(0, -\min(\frac{1-r^T}{1-r} 2^{-t}, 1) \min(2^t f_Q(z) - f_P(z), 0))dz \\
&= \max(1 - \frac{1-r^T}{1-r} 2^{-t}, 0) - \int_z \max(0, -\min(\frac{1-r^T}{1-r} 2^{-t}, 1) \min(2^t f_Q(z) - f_P(z), 0))dz \\
&= \max(1 - \frac{1-r^T}{1-r} 2^{-t}, 0) + \int_z \min(\frac{1-r^T}{1-r} 2^{-t}, 1) \min(2^t f_Q(z) - f_P(z), 0)dz \\
&= \max\left(1 - \frac{1-r^T}{1-r} 2^{-t}, 0\right) + \min\left(\frac{1-r^T}{1-r} 2^{-t}, 1\right) \int_z \min(2^t f_Q(z) - f_P(z), 0)dz \\
&= \max\left(1 - \frac{1-r^T}{1-r} 2^{-t}, 0\right) - \min\left(\frac{1-r^T}{1-r} 2^{-t}, 1\right) \int_z \max(f_P(z) - 2^t f_Q(z), 0)dz \\
&= \max\left(1 - \frac{1-r^T}{1-r} 2^{-t}, 0\right) - \min\left(\frac{1-r^T}{1-r} 2^{-t}, 1\right) \delta \\
&= \max(r^T, \delta) \\
&= \max((1 - (1-\delta)2^{-t})^T, \delta)
\end{aligned}
$$

B Proof of Lemma 8

For every $j \in [m]$ and $z \in \mathcal{I}_j$ we have that

$$\frac{f_{Z_x}(z)}{f_{Q_x}(z)} \leq \frac{M_{x,j}}{\frac{1}{2}\left(\frac{P_{Z_x}(\mathcal{I}_j)}{d(1-\delta_x)} + \frac{1}{D}\right)} \leq \frac{2M_{x,j}}{\left[\frac{P_{Z_x}(\mathcal{I}_j)}{d} + \frac{1}{D}\right]} \leq 2^{t_x}$$

for the choice of t_x in the lemma statement. Hence, $f_{Z_x}(z) - 2^{t_x} f_{Q_x}(z) \leq 0$ for every $z \in \mathcal{I}_j$. Consequently,

$$\begin{aligned}
\mathsf{SD}_{t_x}(P_{Z_x}, Q_x) &= \int \max\left(0, f_{Z_x}(z) - 2^{t_x} f_{Q_x}(z)\right)\, dz \\
&= \int_{\mathcal{R}} \max(0, f_{Z_x}(z) - 2^{t_x} f_{Q_x}(z))\, dz \\
&+ \int_{\overline{\mathcal{R}}} \max(0, f_{Z_x}(z) - 2^{t_x} f_{Q_x}(z))\, dz \\
&\leq 0 + \int_{\overline{\mathcal{R}}} \max(0, f_{Z_x}(z))\, dz \\
&= P_{Z_x}(\overline{\mathcal{R}}) \\
&= \delta_x,
\end{aligned}$$

as desired.

C Proof of Lemma 6

Let $d_1, \ldots, d_{m+1}$ be the images of d. Without loss of generality, we can assume that $d_1 \leq d_2 \leq \ldots \leq d_{m+1}$. For convenience let $d_0 = -\infty$ and $d_{m+2} = +\infty$. Now, if the additive noise N has a probability density function P_N which is radially symmetric and decreasing then on the interval $(\frac{d_i+d_{i-1}}{2}, \frac{d_i+d_{i+1}}{2})$, we have $\max_x P_{Z|X=x}(z) = \varphi(z - d_i)$, $i = 1, \ldots, m+1$. As a consequence,

$$\begin{aligned}
\int_z \max_x p_{Z|X}(z|x)dz &= \sum_{i=1}^{m+1} \mathbb{P}(d_i + N \in (\frac{d_i + d_{i-1}}{2}, \frac{d_i + d_{i+1}}{2})), \\
&= \sum_{i=1}^{m+1} (\Phi(\frac{d_{i+1} - d_i}{2}) - \Phi(\frac{d_{i-1} - d_i}{2})), \\
&= 1 + \sum_{i=1}^{m} (\Phi(\frac{\Delta_i}{2}) - \Phi(-\frac{\Delta_i}{2})), \\
&= 1 + \sum_{i=1}^{m} \mathbb{P}(N \in (\frac{-\Delta_i}{2}, \frac{\Delta_i}{2})).
\end{aligned}$$

Now, since φ is radially symetric and decreasing, the mapping $x \in [0, +\infty) \mapsto \mathbb{P}(N \in [-x, x]) \in [0, 1]$ is concave. As a consequence, for a fixed $\Delta = \sum_{i=1}^{m} \Delta_i$,

$\sum_{i=1}^{m} \mathbb{P}(N \in (\frac{-\Delta_i}{2}, \frac{\Delta_i}{2}))$ is maximised when $\Delta_1 = \ldots = \Delta_m = m^{-1}\Delta$. Indeed, if it was not the case, then by the absurd we would have a maximiser with at least one pair $\Delta_i \neq \Delta_j$. But then by concavity, $2\mathbb{P}(N \in (\frac{-(\Delta_i+\Delta_j)}{4}, \frac{\Delta_i+\Delta_j}{4})) \geq \mathbb{P}(N \in (\frac{-\Delta_i}{2}, \frac{\Delta_i}{2})) + \mathbb{P}(N \in (\frac{-\Delta_j}{2}, \frac{\Delta_j}{2}))$ which is a contradiction. As a consequence,

$$\sum_{i=1}^{m} \mathbb{P}(N \in (\frac{-\Delta_i}{2}, \frac{\Delta_i}{2})) \leq m\mathbb{P}(N \in (\frac{-\Delta}{2m}, \frac{\Delta}{2m})).$$

Finally, by upper bounding the integral by a rectangle,

$$m\mathbb{P}(N \in (\frac{-\Delta}{2m}, \frac{\Delta}{2m})) \leq m\frac{\Delta}{m}\varphi(0) = \frac{\Delta}{\sqrt{2\pi}}.$$

D Proof of Theorem 8

To see that Z is $(\boldsymbol{\mathcal{I}}, \boldsymbol{M}, \delta)$-noisy leakage from X, we invoke Lemma 9 with $f = f_{Z_x}$ and $\mathcal{I} = \mathcal{I}_j$ to conclude that for each interval $\mathcal{I}_j$ and $x \in \mathcal{X}$ we have

$$\max_{z \in \mathcal{I}_j} f_{Z_x}(z) \leq \max\left(\sqrt{2P_{Z_x}(\mathcal{I}_j)\gamma}, 2P_{Z_x}(\mathcal{I}_j)/d\right).$$

Fix $\alpha > 0$. Combining the fact that Z is $(\boldsymbol{\mathcal{I}}, \boldsymbol{M}, \delta)$-noisy leakage from X with Theorem 7 we conclude that Z is $(2\alpha + 3\delta)$-simulatable from

$$\ell = \lceil t + \log\ln(1/\alpha) + \log(D/d) \rceil \tag{7}$$

bits of bounded leakage from X, where

$$t = \max_{x \in \mathcal{X}, j \in [m]} \log\left(\frac{2M_{x,j}}{\frac{P_{Z_x}(\mathcal{I}_j)}{d} + \frac{1}{D}}\right).$$

We now proceed to upper bound t appropriately, using the fact that Z is $(\mathcal{R}, \gamma, \delta)$-Lipschitz-noisy leakage. Define

$$t_{x,j} = \log\left(\frac{2M_{x,j}}{\frac{P_{Z_x}(\mathcal{I}_j)}{d} + \frac{1}{D}}\right).$$

Then, we may write $t = \max_{x \in \mathcal{X}, j \in [m]} t_{x,j}$.

For x and j with $\sqrt{2P_{Z_x}(\mathcal{I}_j)\gamma} \leq 2P_{Z_x}(\mathcal{I}_j)/d$ we have

$$t_{x,j} \leq \log\left(\frac{2P_{Z_x}(\mathcal{I}_j)/d}{\frac{1}{2}\frac{P_{Z_x}(\mathcal{I}_j)}{d} + \frac{1}{2}\frac{1}{D}}\right) \leq \log\left(\frac{2P_{Z_x}(\mathcal{I}_j)/d}{\frac{1}{2}\frac{P_{Z_x}(\mathcal{I}_j)}{d}}\right) \leq 2. \tag{8}$$

On the other hand, for x and j with $\sqrt{2P_{Z_x}(\mathcal{I}_j)\gamma} > 2P_{Z_x}(\mathcal{I}_j)/d$ we have

$$2^{t_{x,j}} \leq \frac{\sqrt{2P_{Z_x}(\mathcal{I}_j)\gamma}}{\frac{1}{2}\frac{P_{Z_x}(\mathcal{I}_j)}{d} + \frac{1}{2}\frac{1}{D}} \leq \min\left(\frac{2d\sqrt{2P_{Z_x}(\mathcal{I}_j)\gamma}}{P_{Z_x}(\mathcal{I}_j)}, 2D\sqrt{2P_{Z_x}(\mathcal{I}_j)\gamma}\right). \tag{9}$$

If $P_{Z_x}(\mathcal{I}_j) \le d/D$, then Equation (9) implies that

$$2^{t_{x,j}} \le 2D\sqrt{2d/D \cdot \gamma} = 2\sqrt{2Dd\gamma}. \tag{10}$$

On the other hand, if $P_{Z_x}(\mathcal{I}_j) \ge d/D$ then

$$2^{t_{x,j}} \le \frac{2d\sqrt{2P_{Z_x}(\mathcal{I}_j)\gamma}}{P_{Z_x}(\mathcal{I}_j)} \le 2\sqrt{2Dd\gamma}. \tag{11}$$

Combining Equations (8), (10) and (11), we conclude that

$$t = \max_{x\in\mathcal{X}, j\in[m]} t_{x,j} \le \max(2, \log(2\sqrt{2Dd\gamma})). \tag{12}$$

Combining Equations (12) and (7) finishes the proof. Furthermore, We get the simulation run time by invoking Theorem 7 with Equation (12).

E Trading Bounded Leakage for Simulation Efficiency in Higher Dimensions

First, let us prove a useful lemma that bounds the supremum of a continuous Lipschitz function (wrt ℓ^∞ norm) over a hypercube in $\mathbb{R}^k$ given the integral of that function over that hypercube.

Lemma 10. *Let $f : \mathbb{R}^k \to \mathbb{R}_{\ge 0}$ be a continuous, non-negative function that is γ-Lipschitz with respect to the ℓ^∞ norm, i.e.,*

$$|f(\mathbf{x}) - f(\mathbf{y})| \le \gamma\|\mathbf{x}-\mathbf{y}\|_\infty \quad \textit{for all } \mathbf{x},\mathbf{y} \in \mathbb{R}^k,$$

Let $C = [0,d]^k \subset \mathbb{R}^k$ be a hypercube with side length $d > 0$ and it satisfies $\int_C f(\mathbf{x})\,d\mathbf{x} = \beta$. If $M := \sup_{\mathbf{x}\in C} f(\mathbf{x})$ we have

$$M \le \max\left(\left((k+1)\,\gamma^k\,\beta\right)^{1/(k+1)}, \quad (k+1)\frac{\beta}{d^k}\right).$$

Proof. Set $r := \frac{M}{\gamma}$. Suppose $f(\mathbf{x}^*) = M$ for some $\mathbf{x}^* \in C$. By the ℓ^∞-Lipschitz condition,

$$f(\mathbf{x}) \ge M - \gamma\|\mathbf{x}-\mathbf{x}^*\|_\infty \quad \text{for all } \mathbf{x} \in C.$$

Define the ℓ^∞ *tent function* centered at $\mathbf{x}^*$:

$$g(\mathbf{x}) = \max\big(M - \gamma\|\mathbf{x}-\mathbf{x}^*\|_\infty,\ 0\big).$$

Its support is the ℓ^∞ ball $B_\infty(\mathbf{x}^*, r) = \{\mathbf{x} : \|\mathbf{x}-\mathbf{x}^*\|_\infty \le r\}$, which is the hypercube $[\mathbf{x}^* - r\mathbf{1}, \mathbf{x}^* + r\mathbf{1}]$ of side $2r$ centered at $\mathbf{x}^*$. Since $f \ge g$ on C and $f \ge 0$,

$$\beta = \int_C f(\mathbf{x})\,d\mathbf{x} \ge \int_C g(\mathbf{x})\,d\mathbf{x} =: A(\mathbf{x}^*).$$

Now, we shall use the Cavalier's principle to estimate $A(x^*)$. First we notice,

$$\{\mathbf{x} : \mathbf{x} \in C \text{ and } g(\mathbf{x}) > t\}$$
$$= \big\{\mathbf{x} : \mathbf{x} \in C \text{ and } \|\mathbf{x} - \mathbf{x}^*\|_\infty < (M-t)/\gamma\big\} = B_\infty\big(\mathbf{x}^*,\, (M-t)/\gamma\big) \cap C$$

By the layer-cake representation,

$$A(\mathbf{x}^*) = \int_0^M \mathrm{Vol}\big(B_\infty(\mathbf{x}^*, \tfrac{M-t}{\gamma}) \cap C\big)\, dt$$

Since $B_\infty(\mathbf{x}^*, \rho) \cap C$ is an axis-aligned box with side lengths $\ell_i(x_i^*, \rho) = \min(x_i^* + \rho, d) - \max(x_i^* - \rho, 0)$, and each ℓ_i is minimized at the endpoints of $[0, d]$, the volume $\mathrm{Vol}(B_\infty(\mathbf{x}^*, \rho) \cap C)$ and hence $A(\mathbf{x}^*)$ is minimized at a corner of C for every $\rho > 0$. Therefore, in what follows we work with the corner $\mathbf{x}^* = \mathbf{0}$, so $g(\mathbf{x}) = \max(M - \gamma\|\mathbf{x}\|_\infty, 0)$. We have

$$\int_{\mathbb{R}^k} g(\mathbf{x})\, d\mathbf{x}$$
$$= \int_0^M \mathrm{Vol}\big(B_\infty(\mathbf{0}, \tfrac{M-t}{\gamma})\big)\, dt = \int_0^M \left(\frac{2(M-t)}{\gamma}\right)^k dt = \frac{2^k M^{k+1}}{(k+1)\gamma^k}$$

Case 1: Small Radius ($r \le d$): At the corner $\mathbf{x}^* = \mathbf{0}$, the support of g is $B_\infty(\mathbf{0}, r) = [-r, r]^k$. When $r \le d$, this ball is contained in $[-d, d]^k$, so it does not reach the far faces $\{x_i = d\}$ of C. The intersection $B_\infty(\mathbf{0}, r) \cap [0, d]^k$ therefore equals $B_\infty(\mathbf{0}, r) \cap [0, \infty)^k = [0, r]^k$, the corner hypercube of side r.

Since $g(\mathbf{x}) = \max(M - \gamma\|\mathbf{x}\|_\infty, 0)$ is symmetric under sign-flips of coordinates, and $B_\infty(\mathbf{0}, r)$ is likewise symmetric, the 2^k orthants each contribute equally. The positive orthant $[0, \infty)^k$ thus contributes exactly $1/2^k$ of the integral over $B_\infty(\mathbf{0}, r)$:

$$A(\mathbf{0}) = \int_{[0,r]^k} g(\mathbf{x})\, d\mathbf{x} = \frac{1}{2^k} \int_{\mathbb{R}^k} g(\mathbf{x})\, d\mathbf{x} = \frac{1}{2^k} \cdot \frac{2^k M^{k+1}}{(k+1)\gamma^k} = \frac{M^{k+1}}{(k+1)\gamma^k}.$$

Since $\beta \ge A(\mathbf{0})$:

$$\beta \ge \frac{M^{k+1}}{(k+1)\gamma^k} \implies M \le \big((k+1)\,\gamma^k\,\beta\big)^{1/(k+1)}.$$

Case 2: General Radius: For any $r > 0$, we have $\max(t, 0) \ge t$.

$$g(\mathbf{x}) = \max(M - \gamma\|\mathbf{x}\|_\infty,\, 0) \ge M - \gamma\|\mathbf{x}\|_\infty.$$

Integrating over $C = [0, d]^k$:

$$A(\mathbf{0}) \ge \int_{[0,d]^k} \big(M - \gamma\|\mathbf{x}\|_\infty\big)\, d\mathbf{x} = M\, d^k - \gamma \int_{[0,d]^k} \|\mathbf{x}\|_\infty\, d\mathbf{x} = M\, d^k - \frac{k\,\gamma\, d^{k+1}}{k+1}$$

Thus,

$$\beta \ge A(\mathbf{0}) \ge M\, d^k - \frac{k\,\gamma\, d^{k+1}}{k+1} \implies M \le \frac{\beta}{d^k} + \frac{k\,\gamma\, d}{k+1}$$

Additionally, in case of $r = \frac{M}{\gamma} > d$ we have $M < (k+1)\frac{\beta}{d^k}$.

Finally, combining case 1 and 2 we get

$$M \leq \max\left(\left((k+1)\,\gamma^k\,\beta \right)^{1/(k+1)}, \quad (k+1)\frac{\beta}{d^k} \right).$$

□

Remark 4. We note, if we assume the Lipschitz condition on the pdf f_{Z_x} of the k dimensional leakage Z in ℓ^2 norm (instead of ℓ^∞ norm), which may be easy to test for practical purposes, we by default, recover the ℓ^∞ Lipschitz condition with a $\sqrt{k}$ factor blow-up in the Lipschitz constant.

We are now ready to extend the framework of Definition 5 and Algorithm 1 where the leakage Z comes from $\mathbb{R}^k$ instead of $\mathbb{R}$ assuming an additional Lipschitz structure on f_{Z_x}. Lemma 8 from above evolves in the following way

Lemma 11. *Let $\mathcal{R}$ be a hypercube of each side length D in $\mathbb{R}^k$. Set $m = \lceil D/d \rceil$ and let $\boldsymbol{\mathcal{I}} = (\mathcal{I}_{\boldsymbol{j}})_{\boldsymbol{j}\in[m]^k}$ be a partition of $\mathcal{R}$ into disjoint hypercubes each of side length $d < D$. Suppose that Z is an $(\boldsymbol{\mathcal{I}}, \boldsymbol{M}, \delta)$-noisy leakage from X (where $\boldsymbol{M} = (M_{x,\boldsymbol{j}})_{x\in\mathcal{X}, j\in[m]^k} \in \mathbb{R}^{|\mathcal{X}|\times m^k}$) in the following sense*

- *$f_{Z_x}(z) \leq M_{x,\boldsymbol{j}}$ for all $z \in \mathcal{I}_{\boldsymbol{j}}$, $\boldsymbol{j} \in [m]^k$, and $x \in \mathcal{X}$,*
- *$\mathbb{E}_{x\sim P_X}\left[P_{Z_x}(\bigcup_{\boldsymbol{j}\in[m]^k} \mathcal{I}_{\boldsymbol{j}})\right] \geq 1 - \delta$.*

For a given $x \in \mathcal{X}$ with $\delta_x = 1 - P_{Z_x}(\mathcal{R}) < 1$, consider a distribution Q_x with pdf f_{Q_x} satisfying

$$f_{Q_x}(z) = \begin{cases} \frac{1}{2}\left(\frac{P_{Z_x}(\mathcal{I}_{\boldsymbol{j}})}{d^k(1-\delta_x)} + \frac{1}{D^k} \right), & \text{if } z \in \mathcal{I}_{\boldsymbol{j}}, \\ 0, & \text{if } z \notin \mathcal{R}. \end{cases}$$

Then, for

$$t_x = \max_{\boldsymbol{j}\in[m]^k} \log\left(2M_{x,\boldsymbol{j}} \cdot \left[P_{Z_x}(\mathcal{I}_{\boldsymbol{j}}) \cdot d^{-k} + D^{-k} \right]^{-1} \right)$$

it holds that $\mathsf{SD}_{t_x}(P_{Z_x};\, Q_x) \leq \delta_x$.

Proof. Exactly analogous to the proof of Lemma 8 with additionally taking into account the partition in dimension k instead of 1. □

In a similar way, we extend Algorithm 1 in the following ways

- All the leakages are now element of $\mathbb{R}^k$ instead of $\mathbb{R}$ i.e. we have $(\boldsymbol{z_i})_{i\in[T]}$ to start with.
- All the partition indices are now elemnts of $[m]^k$ instead of $[m]$.
-
- In line 3, sample $\boldsymbol{j_0}$ uniformly at random from $[m]^k$. In line 8. set $\boldsymbol{j_1}$ to be the unique vector representing the hypercube index such that $y \in \mathcal{I}_{\boldsymbol{j_1}}$.
- In line 17, the translated leakage sample becomes $\boldsymbol{z_i}' \leftarrow \boldsymbol{z_i} + (\boldsymbol{a} + (\boldsymbol{j} - \boldsymbol{1}).d)$ where $\boldsymbol{1}$ is the all 1 vector.

Analogously, in the extended version of Theorem 7 we have

- $t := \max_{x\in\mathcal{X}} t_x$
- $\ell = \lceil t + \log\ln(1/\alpha) + k\log(D/d) \rceil$
- Simulation time (assuming oracle accesses) remains unchanged in terms of t.

Comparison of bounded leakage and simulation efficiency in $\mathbb{R}^k$ obtained via the baseline simulation due to Theorem 2 and efficient simulation due to the extended version of Algorithm 1: First we assume that the pdf of the leakage f_{Z_x} satifies the ℓ^∞ γ-Lipschitz property of Lemma 10. We have

$$t = \max_{x\in\mathcal{X},\boldsymbol{j}\in[m]^k} \log\left(\frac{2M_{x,\boldsymbol{j}}}{\frac{P_{Z_x}(\mathcal{I}_{\boldsymbol{j}})}{d^k}+\frac{1}{D^k}}\right).$$

We now proceed to upper bound t appropriately, using the Lipschitz property. Define

$$t_{x,\boldsymbol{j}} = \log\left(\frac{2M_{x,\boldsymbol{j}}}{\frac{P_{Z_x}(\mathcal{I}_j)}{d^k}+\frac{1}{D^k}}\right).$$

Then, we may write $t = \max_{x\in\mathcal{X},\boldsymbol{j}\in[m]^k} t_{x,\boldsymbol{j}}$. Also, set $\beta := P_{Z_x}(\mathcal{I}_j)$

For x and j with $\big((k+1)\,\gamma^k\,\beta\big)^{1/(k+1)} \leq \quad (k+1)\frac{\beta}{d^k}$ we have

$$t_{x,\boldsymbol{j}} \leq \log\left(\frac{2(k+1)\beta/d^k}{\beta/d^k+1/D^k}\right) \leq \log(2(k+1))$$

On the other hand, for x and $\boldsymbol{j}$ with $\big((k+1)\,\gamma^k\,\beta\big)^{1/(k+1)} > \quad (k+1)\frac{\beta}{d^k}$ we have

$$\begin{aligned} t_{x,\boldsymbol{j}} &\leq \log\left(\frac{2\big((k+1)\,\gamma^k\,\beta\big)^{1/(k+1)}}{\frac{\beta}{d^k}+\frac{1}{D^k}}\right) \\ &\leq \log\Big(2\big((k+1)\,\gamma^k\,\beta\big)^{1/(k+1)}.min(\beta/d^k, D^k)\Big) \end{aligned}$$

Considering both the cases $\beta < (d/D)^k$ and $\beta \geq (d/D)^k$ we have, in the both the cases

$$t_{x,\boldsymbol{j}} \leq \log\left(2.D^k\left[\frac{(k+1)\gamma^k d^k}{D^k}\right]^{1/(k+1)}\right)$$

we conclude that

$$t = \max_{x\in\mathcal{X},\boldsymbol{j}\in[m]^k} t_{x,\boldsymbol{j}} \leq \max\left[\log(2(k+1)), \log\left(2.D^k\left[\frac{(k+1)\gamma^k d^k}{D^k}\right]^{1/(k+1)}\right)\right]$$

□

Now, for the leakage model described in Lemma 10, if we use the baseline simulation Theorem 2, we have $\mu(\mathcal{R}) = D^k, M = \big((k+1)\gamma^k\big)^{1/(k+1)}$, $T = \mu(\mathcal{R}).M.\ln(1/\varepsilon)/(1-\delta)$. For ε simulation error, We need $\ell_1 = \lceil \log T \rceil$ bits of bounded leakage and the simulation runtime is $\mathcal{O}(T)$ assuming oracle accesses.

Using the extended version of the simulation Algorithm 1, we need $\ell_2 =$

$\lceil t + \log\ln(2/(\varepsilon - 3\delta)) + k\log(D/d)\rceil$ bits of bounded leakage with simulator runtime of $\widetilde{\mathcal{O}}(2^t)$.

This entails an extra bounded leakage of $\ell_2 - \ell_1 = \frac{k^2}{k+1}\log(D/d)$ bits to gain a simulation efficiency of $(D/d)^{k/(k+1)}$ ignoring the terms involving ε, δ.

F Applications

We explore applications of simulation theorems for noisy leakage from bounded leakage to obtaining leakage-resilient cryptographic primitives in the *computational* setting. Following the literature [33], we distinguish between applications in which either the memory or the computation leaks information.

F.1 Leakage from Computation

A prominent line of research in the setting of computation leakage aims at constructing specific primitives which exploit the existence of a leak-free memory: the values that are stored in the memory cannot leak, but whenever some of these values are moved from the memory to the CPU in order to perform some computations, they become subject to leakage. This assumption was introduced for the first time by Micali and Reyzin [40] and became known as the axiom that only computation leaks information.

Below, for concreteness, we focus on the case of leakage-resilient stream ciphers. A stream cipher is a function $\mathsf{sc} : \{0,1\}^n \to \{0,1\}^n \times \{0,1\}^m$ that for every secret state $X_0 \in \{0,1\}^n$ defines a sequence of $q+1$ outputs $Y_1, \ldots, Y_{q+1} \in \{0,1\}^m$, for $q \in \mathbb{N}$, that are determined recursively as $(X_{i+1}, Y_{i+1}) = \mathsf{sc}(X_i)$. The standard security notion of a stream cipher requires that, for random $X_0 = U_n$, the outputs $Y_1, \ldots, Y_{q+1}$ are pseudorandom. Leakage resilience, on the other hand, requires that Y_{q+1} is pseudorandom even given $Y_1, \ldots, Y_q$ along with the leakages $Z_1, \ldots, Z_q$, where $Z_i = f_i(X_i^+)$ and X_i^+ is the part of the state that is needed in the i-th round of the computation.

The definition below formalizes the latter guarantee. Let X, Y be discrete random variables with support $\mathcal{Z}$. Throughout this section, we write

$$\mathsf{CD}_t(X;Y) = \max_{\mathsf{D}:\mathcal{Z}\to\{0,1\}} \left| \Pr_{z\sim P_X}[\mathsf{D}(z)=1] - \Pr_{z\sim P_Y}[\mathsf{D}(z)=1] \right|,$$

where the maximum is over all distinguishing algorithms D running in time t.

Remark 5. In Sections 3 to 5 we consider real-valued leakages. To stay within standard computation models, in this section and in Appendix F.2 we assume that algorithms receive as input a discrete representation of the leakage. Since our simulation guarantees in other sections are always obtained with respect to the statistical distance (e.g., see Definition 2 and Theorem 2) and any discretization process can be seen as post-processing of the leakage, the discrete representations of the true leakage and the simulated leakage are still statistically close (with

the same parameters) even when the secret X is revealed to the distinguisher. We will use this fact throughout this section.

Definition 7 (Leakage-resilient stream cipher). *Let* $\mathsf{sc} : \{0,1\}^n \to \{0,1\}^n \times \{0,1\}^m$ *be a stream cipher. We say that* sc *is* $(\mathcal{F}, \varepsilon, t, q)$-noisy leakage-resilient *if for* X_0 *uniformly distributed over* $\{0,1\}^n$ *and for every sequence of adaptively chosen leakage functions* $f_1, \ldots, f_q \in \mathcal{F}$ *it holds that*

$$\mathsf{CD}_t((Y_{q+1}, Y_1, \ldots, Y_q, Z_1, \ldots, Z_q); (U_m, Y_1, \ldots, Y_q, Z_1, \ldots, Z_q)) \leq \varepsilon,$$

where for each $i \in [q]$ *we have* $(X_{i+1}, Y_{i+1}) = \mathsf{sc}(X_i)$ *and* $Z_i = f_i(X_i^+)$.

In case $\mathcal{F} = \mathcal{G}$ *coincides with the set of* ℓ*-bounded leakage functions, we simply say that* sc *is* $(\ell, \varepsilon, t, q)$-bounded leakage-resilient*. Moreover, in case the sequence of leakage functions is chosen non-adaptively, we say that* sc *is* non-adaptive leakage-resilient*.*

The first construction of a bounded leakage-resilient stream cipher was proposed by Dziembowski and Pietrzak [23] and later simplified by Pietrzak [44]. In these constructions, the initial state consists of three n-bit strings $X_0 = (K_0, K_1, Y_0)$, where Y_0 can be made public, and at round $i \in [q]$ the stream cipher sc takes as input state $X_i = (K_{i-1}, K_i, Y_{i-1})$ and uses $X_i^+ = (K_{i-1}, Y_{i-1})$ to produce $(K_{i+1}, Y_i) = \mathsf{prf}(K_{i-1}, Y_{i-1})$, where prf is any (weak) pseudorandom function. The alternating structure avoids the so-called leakage pre-computation attack, in which the adversary at round $i, i+1, \ldots$ leaks ℓ bits of a future state at round $j \gg i$, thus eventually revealing the entire state X_j and breaking security. Yu, Standaert, Pereira, and Yung [53] simplified the construction even further, assuming the leakage is both bounded and non-adaptive. In their construction, the initial state consists of three n-bit strings $X_0 = (K_0, R_0, R_1)$, where both R_0 and R_1 can be made public, and at round $i \in [q]$ the stream cipher takes as input state $X_i = (K_i, R_0, R_1)$ and uses $X_i^+ = (K_i, R_{i \bmod 2})$ to produce $(K_{i+1}, Y_{i+1}) = \mathsf{prf}(K_i, R_{i \bmod 2})$, where prf is any (weak) pseudorandom function. The pre-computation attack is avoided by the assumption that the leakage is non-adaptive and thus cannot depend on R_0, R_1. Unfortunately, this construction is not provably secure unless one assumes an arbitrarily long sequence of uniformly random values $R_1, R_2, \ldots$ [26]. Yu and Standaert [52] get around this limitation by using a single public value R that is expanded using a PRG. However, this construction can only be proven secure in `minicrypt` (i.e., assuming that there is no black-box construction of public-key encryption from one-way functions).

The theorem below shows that every stream cipher that is secure against bounded non-adaptive[11] leakage is also secure against noisy non-adaptive leakage via any family $\mathcal{F}$, so long as $\mathcal{F}$ is simulatable from bounded leakage with efficient simulation and small error. In order to make precise statements about the running time of the leakage simulation, we will simply write that $\mathcal{F}$ is (ε, t)-simulatable from bounded leakage, where t represents the simulator's running time. For simplicity, we count the complexity of the simulator's oracle calls as

[11] The results in Appendix F.2 consider the case of adaptive leakage.

$O(1)$ overhead; as we discussed, this is a reasonable assumption for many realistic leakage families (see Remark 1).[12]

Theorem 9. *Let $\mathcal{F}$ be $(\varepsilon_{\mathrm{sim}}, t_{\mathrm{sim}})$-simulatable from ℓ bits of bounded leakage. Then, for any stream cipher $\mathsf{sc} : \{0,1\}^n \to \{0,1\}^n \times \{0,1\}^m$ that is non-adaptive $(\ell, \varepsilon_{\mathrm{sc}}, t_{\mathrm{sc}}, q)$-bounded leakage-resilient is also non-adaptive $(\mathcal{F}, \varepsilon, t, q)$-noisy leakage-resilient for $\varepsilon = 2q \cdot \varepsilon_{\mathrm{sim}} + \varepsilon_{\mathrm{sc}}$ and $t_{\mathrm{sc}} = q \cdot t_{\mathrm{sim}} + t$.*

Proof. Fix any $q \in \mathbb{N}$ and any sequence of leakage functions $f_1, \ldots, f_q \in \mathcal{F}$. We start with the initial distribution:

$$V = (Y_{q+1}, Y_1, \ldots, Y_q, Z_1, \ldots, Z_q),$$

where $X_0 = U_n$, and for each $i \in [q]$ we have $(X_{i+1}, Y_{i+1}) = \mathsf{sc}(X_i)$ and $Z_i = f_i(X_i^+)$. Next, we modify the distribution V using the simulator Sim of Definition 2 (see Remark 2):

$$\tilde{V} = (Y_{q+1}, Y_1, \ldots, Y_q, \tilde{Z}_1, \ldots, \tilde{Z}_q),$$

where, for each $i \in [q]$, we set $\tilde{Z}_i = \mathsf{Sim}_2(g_i(X_i^+))$ for the ℓ-bounded leakage function g_i returned by $\mathsf{Sim}_1(f_i)$. The lemma below says that the distributions V and W are computationally indistinguishable.

Lemma 12. $\mathsf{CD}_{t_{\mathrm{sc}}}(V; \tilde{V}) \leq q \cdot \varepsilon_{\mathrm{sim}}$.

Proof. The proof uses a hybrid argument. Let H_i be the distribution defined as $(Y_{q+1}, Y_1, \ldots, Y_q, \tilde{Z}_1, \ldots, \tilde{Z}_i, Z_{i+1}, \ldots, Z_q)$ where the leakages are as defined above. Note that $H_0 = V$ and $H_q = \tilde{V}$. Thus, it suffices to show that for each $i \in [0, q-1]$ we have that H_i and H_{i+1} are indistinguishable. Fix an $i \in [0, q-1]$ and any sequence of leakage functions $f_1, \ldots, f_q \in \mathcal{F}$. By contradiction, assume that there is a distinguisher D with running time t_{sc} telling apart H_i and H_{i+1} with probability greater than $\varepsilon_{\mathrm{sim}}$. We build a distinguisher[13] D' that contradicts Definition 2 w.r.t. target distribution $\hat{X} = X_0 = U_n$.

- Let $\hat{f} \in \mathcal{F}$ be the leakage function that for each $j \leq i$ determines $(X_{j+1}, Y_{j+1}) = \mathsf{sc}(X_j)$ and then outputs $f_{i+1}(X_{i+1}^+)$. At the onset, D' forwards the leakage function $\hat{f}$ to its challenger and receives back a pair $(\hat{X}, \hat{Z})$.
- Next, for each $j \in [q]$, D' computes $(X_{j+1}, Y_{j+1}) = \mathsf{sc}(X_j)$ and sets $\tilde{Z}_j = \mathsf{Sim}_2(\mathsf{Sim}_1(f_j))$ for each $j \leq i$ and $Z = f_j(X_j^+)$ for each $j \geq i+1$.
- Finally, D' passes $(Y_{q+1}, Y_1, \ldots, Y_q, \tilde{Z}_1, \ldots, \tilde{Z}_i, \hat{Z}, Z_{i+2}, \ldots, Z_q)$ to D and outputs whatever D outputs.

We note that in case $\hat{Z}$ equals the output of $\mathsf{Sim}_2(\mathsf{Sim}_1(f_{i+1}))$, the view of D is identical to that of H_{i+1}; on the other hand, in case $\hat{Z}$ equals the output of $f_{i+1}(X_{i+1}^+)$, the view of D is identical to that of H_i. Hence, the statement of the lemma follows by Definition 2. □

[12] We remark that most prior works on leakage-resilient stream ciphers consider a non-uniform model of computation where distinguishers are circuits of bounded size. These results can be translated to the uniform model of computation considered here.

[13] While D' will be efficient, its running time does not matter as Definition 2 guarantees statistical (rather than computational) distance.

Next, we change the distribution $\tilde{V}$ by replacing Y_{q+1} with uniform U_m:

$$\tilde{W} = (U_m, Y_1, \ldots, Y_q, \tilde{Z}_1, \ldots, \tilde{Z}_q).$$

Lemma 13. $\mathsf{CD}_t(\tilde{V}; \tilde{W}) \leq \varepsilon_{\mathrm{sc}}$.

Proof. By contradiction, assume that there exists a distinguisher with running time t telling apart the distributions $\tilde{V}$ and $\tilde{W}$ with probability larger than $\varepsilon_{\mathrm{sc}}$. We build a distinguisher D' with running time $q \cdot t_{\mathrm{sim}} + t = t_{\mathrm{sc}}$ that contradicts non-adaptive $(\ell, \varepsilon_{\mathrm{sc}}, t_{\mathrm{sc}}, q)$-bounded leakage resilience of the stream cipher. The distinguisher D' proceeds as follows.

- At the onset, D' receives from D the sequence of leakage functions $f_1, \ldots, f_q \in \mathcal{F}$. For each $i \in [q]$, the distinguisher D' runs $\mathsf{Sim}_1(f_i)$ and obtains a corresponding ℓ-bounded leakage function $g_i \in \mathcal{G}$.
- For each $i \in [q]$, let $\hat{g}_i \in \mathcal{G}$ be the leakage function that for each $j \leq i$ determines $(X_{j+1}, Y_{j+1}) = \mathsf{sc}(X_j)$ and then outputs $g_i(X_i^+)$. The distinguisher D' forwards $(\hat{g}_1, \ldots, \hat{g}_q)$ to its challenger for target $\hat{X} = X_0 = U_n$ and receives back $(\hat{Y}, Y_1, \ldots, Y_q, \hat{Z}_1, \ldots, \hat{Z}_q)$.
- Finally, D' passes $(\hat{Y}, Y_1, \ldots, Y_q, \mathsf{Sim}_2(\hat{Z}_1), \ldots, \mathsf{Sim}_2(\hat{Z}_q)$ to D and outputs whatever D outputs.

We note that D' perfectly simulates the distribution of the leakage values in both $\tilde{V}$ and $\tilde{W}$. Furthermore, the view of D is either identical to $\tilde{V}$ or to $\tilde{W}$ depending on $\hat{Y} = Y_{q+1}$ or $\hat{Y} = U_m$. Hence, the statement of the lemma follows by Definition 7. □

Finally, let $W = (U_m, Y_1, \ldots, Y_q, Z_1, \ldots, Z_q)$ be the distribution that is identical to $\tilde{W}$ except that we replace the simulated leakages $\tilde{Z}_i = \mathsf{Sim}_2(\mathsf{Sim}_1(f_i))$ with the real leakages $Z_i = f_i(X_i^+)$. An argument almost identical to that used in the proof of Lemma 12 shows that $\mathsf{CD}_{t_{\mathrm{sc}}}(\tilde{W}; W) \leq q \cdot \varepsilon_{\mathrm{sim}}$. Theorem 9 follows. □

By combining Theorem 9 with Corollary 1, for example, we obtain noisy leakage-resilient stream ciphers in the practically-relevant settings in which the leakage outputs are Gaussian-noisy. We omit to count the overhead due to oracle access in the leakage simulation.

Corollary 3. *Let $\mathcal{F} = \{f : \mathcal{X} \to \mathcal{Z}\}$ be the family of leakage functions specified as $Z = f(X) = d(X) + \sigma\mathcal{N}(0,1)$, and let $w = \max_x d(x) - \min_x d(x)$. Then, for $\varepsilon_{\mathrm{sim}} > 0$, any stream cipher $\mathsf{sc} : \{0,1\}^n \to \{0,1\}^n \times \{0,1\}^m$ that is non-adaptive $(\ell, \varepsilon_{\mathrm{sc}}, t_{\mathrm{sc}}, q)$-bounded leakage-resilient is also non-adaptive $(\mathcal{F}, 2q \cdot \varepsilon_{\mathrm{sim}} + \varepsilon_{\mathrm{sc}}, t_{\mathrm{sc}} - q \cdot \mathcal{O}(T), q)$-noisy leakage-resilient, for*

$$\ell = \lceil \log_2(T) \rceil \quad \textit{and} \quad T = \left\lceil \frac{\ln \frac{1}{\varepsilon}}{-\ln\left(1 - \frac{\sqrt{2\pi}(1-\varepsilon)}{\frac{w}{\sigma} + 2\Phi^{-1}(1-\frac{\varepsilon}{2})}\right)} \right\rceil.$$

F.2 Leakage from Memory

In the memory leakage model, the attacker is allowed to obtain some leakage from the memory of a cryptosystem, which usually contains the secret key $X \in \mathcal{X}$. Most of the results consider ℓ-bounded leakage, which basically means that the leakage consists of at most ℓ bits for some parameter $\ell \in \mathbb{N}$ (shorter than the length of X). In the subsections below, we study memory leakage resilience of arbitrary cryptographic primitives and protocols with either game-based or simulation-based security.

Let $X = \{X_\lambda\}_{\lambda\in\mathbb{N}}$ and $Y = \{Y_\lambda\}_{\lambda\in\mathbb{N}}$ be distribution ensembles. Throughout this section, we write $X \approx_c Y$ to denote that X and Y are computationally indistinguishable, namely, for all probabilistic polynomial-time (PPT) distinguishers D it holds that $|\Pr[\mathsf{D}(X_\lambda) = 1] - \Pr[\mathsf{D}(Y_\lambda) = 1]| \leq \mathrm{negl}(\lambda)$, where $\mathrm{negl}(\lambda)$ denotes a negligible function in the security parameter $\lambda \in \mathbb{N}$ (i.e., $\mathrm{negl}(\lambda) = \lambda^{-\omega(1)}$). Sometimes, we also write $X \approx_s Y$ to denote that the statistical distance between X and Y is negligible. We write $\mathrm{poly}(\lambda)$ to denote an unspecified polynomial in the security parameter $\lambda \in \mathbb{N}$.

As noted in Appendix F (see Remark 5), to stay within standard computation models, even though in the rest of the paper we consider leakage functions outputting real values, in this section we assume that algorithms receive as input a discrete representation of the leakage.

Game-based Style. Let Π be a cryptosystem; we do not make any assumptions about the algorithms underlying Π, except that we assume that there exists a PPT algorithm $\mathsf{Gen}(1^\lambda)$ taking as input some security parameter $\lambda \in \mathbb{N}$ and outputting a secret key $X \in \mathcal{X}$ along with public information $Y \in \mathcal{Y}$ (possibly empty). Furthermore, we assume that there exists an optional PPT algorithm $\mathsf{Update}(X)$ taking as input a secret key $X \in \mathcal{X}$ and outputting a uniformly random secret key $X' \in \mathcal{X}$ without changing the public key Y. We model leakage resilience of Π via an interactive game between a PPT attacker A and a PPT challenger C (whose behavior is dictated by the security property for Π). The game proceeds for q rounds, where $q = \mathrm{poly}(\lambda)$, and where each round $i \in [q]$ is structured as follows:

- **Key update / generation.** In the first round $i = 1$, the challenger $\mathsf{C}(1^\lambda)$ runs $(X, Y) \leftarrow\!\!\$\ \mathsf{Gen}(1^\lambda)$ and forwards Y to $\mathsf{A}(1^\lambda)$. In the generic i-th round, with $i > 1$, the challenger $\mathsf{C}(1^\lambda)$ runs $X_{i+1} \leftarrow\!\!\$\ \mathsf{Update}(X_i)$.
- **Leakage.** The attacker $\mathsf{A}(1^\lambda)$ outputs an efficiently computable leakage function $f_i : \mathcal{X} \to \mathcal{Z}$ and obtains leakage $Z_i = f_i(X_i)$ from $\mathsf{C}(1^\lambda)$, where $f_i \in \mathcal{F}(X_i)$.
- **Playing the game.** The rest of the interaction between $\mathsf{A}(1^\lambda)$ and $\mathsf{C}(1^\lambda)$ is completely arbitrary. At the end of round q, the challenger $\mathsf{C}(1^\lambda)$ outputs a bit b where $b = 1$ indicates that $\mathsf{A}(1^\lambda)$ has won the game.

Following [19], we consider two kind of games:

- For *unpredictability* games, we define the advantage of attacker $\mathsf{A}(1^\lambda)$ against challenger $\mathsf{C}(1^\lambda)$ fixed by Π as $\Pr[b = 1]$ in the above game.

– For *indistinguishability* games, we define the advantage of attacker $\mathsf{A}(1^\lambda)$ against challenger $\mathsf{C}(1^\lambda)$ fixed by Π as $|\Pr[b = 1] - 1/2|$ in the above game.

Definition 8 (Game-based leakage resilience). *We say that Π is $(\mathcal{F}, q)$-noisy leakage-resilient if for all PPT adversaries A, the advantage of $\mathsf{A}(1^\lambda)$ against challenger $\mathsf{C}(1^\lambda)$ fixed by Π is negligible in λ, where in each round $i \in [q]$ the leakage function f_i belongs to the set $\mathcal{F}(X_i)$ of leakage from X_i. In case $\mathcal{F} = \mathcal{G}$ coincides with the set of ℓ-bounded leakage functions, we simply say that Π is (ℓ, q)-bounded leakage-resilient.*

When $q = 1$ and $\mathcal{F} = \mathcal{G}$, the above definition captures several cryptosystems with bounded leakage resilience, including public-key encryption [1, 41, 17, 29], digital signatures [34, 25, 17, 27, 12, 42, 15, 24], and identity-based encryption [37]. When $q = \text{poly}(\lambda)$ and $\mathcal{F} = \mathcal{G}$, we obtain instead cryptosystems with continual leakage resilience [13, 16, 36, 18].

The theorem below says that any cryptosystem that is secure in the presence of bounded leakage (as defined above) is also secure in the presence of noisy leakage, so long as we can simulate the noisy leakage with bounded leakage efficiently and with negligible simulation error.

Theorem 10. *Let $\mathcal{F}$ be $(\varepsilon_{\text{sim}}, t_{\text{sim}})$-simulatable from ℓ bits of bounded leakage for $t_{\text{sim}} = \text{poly}(\lambda)$ and $\varepsilon_{\text{sim}} = \text{negl}(\lambda)$. Then, for any cryptosystem Π, if Π is (ℓ, q)-bounded leakage-resilient then Π is also $(\mathcal{F}, q)$-noisy leakage-resilient for any $q = \text{poly}(\lambda)$.*

Proof. We consider a sequence of $q + 1$ hybrid experiments $\mathsf{Hyb}_i(\lambda)$, for $i \in [0, q]$, specified as follows:

– Experiment $\mathsf{Hyb}_i(\lambda)$ proceeds identically to the original game defining $(\mathcal{F}, q)$-noisy leakage resilience of Π, except for the way it handles leakage queries $f_j \in \mathcal{F}(X_j)$. In particular:
  - If $j \leq i$, the challenger upon input f_j runs the leakage simulator of Definition 2. By Remark 2, we think of such a simulator as being made of two algorithms $(\mathsf{Sim}_1, \mathsf{Sim}_2)$; algorithm Sim_1 takes as input f_j and outputs a single ℓ-bounded leakage query g_j, whereas algorithm Sim_2 takes as input $g_j(X_j)$ and produces the simulated leakage $\tilde{Z}_j$ which is then returned to the adversary.
  - If $j > i$, the challenger upon input f_j returns $Z_j = f_j(X_j)$.

Note that $\mathsf{Hyb}_0(\lambda)$ is the original game defining noisy leakage resilience of Π; the lemma below implies that this experiment is statistically close to the last experiment $\mathsf{Hyb}_q(\lambda)$, in which all leakage queries are answered using the simulator $(\mathsf{Sim}_1, \mathsf{Sim}_2)$ of Definition 2.

Lemma 14. *For each $i \in [q]$, we have that $\{\mathsf{Hyb}_i(\lambda)\}_{\lambda \in \mathbb{N}} \approx_s \{\mathsf{Hyb}_{i-1}(\lambda)\}_{\lambda \in \mathbb{N}}$.*

Proof. Fix an $i \in [q]$. By contradiction, assume that there is an unbounded distinguisher D and a polynomial $p(\lambda) = \text{poly}(\lambda)$ such that:

$$\left|\Pr\left[\mathsf{D}(\mathsf{Hyb}_i(\lambda)) = 1\right] - \Pr\left[\mathsf{D}(\mathsf{Hyb}_{i-1}(\lambda) = 1)\right]\right| \geq 1/p(\lambda).$$

We build an unbounded distinguisher D' that contradicts Definition 2 w.r.t. a target distribution $\hat{X}$ which is specified in the reduction below.

- At the onset, D' samples $(X_1, Y) \leftarrow\!\!\$\ \mathsf{Gen}(1^\lambda)$ and returns Y to D.
- Next, for each $j \in [q]$, D' answers leakage queries f_j from D as follows:
  - If $j < i$, the distinguisher D' returns the value $\tilde{Z}_j = \mathsf{Sim}_2(g_j)$, where g_j is the ℓ-bounded leakage function determined by $\mathsf{Sim}_1(f_j)$, and later sets $X_{j+1} \leftarrow\!\!\$\ \mathsf{Update}(X_j)$.
  - If $j = i$, the distinguisher D' forwards f_j to its challenger for target $\hat{X} = \mathsf{Update}(X_{i-1})$ and receives back a pair $(\hat{X}, \hat{Z})$. The value $\hat{Z}$ is then returned to D. Furthermore, D' sets $X_{i+1} \leftarrow\!\!\$\ \mathsf{Update}(\hat{X})$.
  - If $j > i$, the distinguisher D' returns $Z_j = f_j(X_j)$ and later sets $X_{j+1} \leftarrow\!\!\$\ \mathsf{Update}(X_j)$.
- The remainder of the interaction between D and its challenger (after the leakage) can be perfectly simulated by D' as it knows X_j for each $j \in [q]$.

We note that in case $\hat{Z}$ equals the output of $\mathsf{Sim}_2(\mathsf{Sim}_1(f_j))$, the view of D is identical to that of $\mathsf{Hyb}_i(\lambda)$; on the other hand, in case $\hat{Z}$ equals the output of $f_j(\hat{X})$, the view of D is identical to that of $\mathsf{Hyb}_{i-1}(\lambda)$. Hence, the statement of the lemma follows by Definition 2 and by the fact that $\varepsilon_{\mathrm{sim}} = \mathrm{negl}(\lambda)$. □

Next, assume by contradiction that there exists a PPT adversary A that wins in $\mathsf{Hyb}_0(\lambda)$ with non-negligible probability $\geq 1/p(\lambda)$ for $p(\lambda) = \mathrm{poly}(\lambda)$. The above lemma implies that the same PPT adversary A wins in $\mathsf{Hyb}_q(\lambda)$ with probability $\geq 1/p(\lambda) - \mathrm{negl}(\lambda) \geq 1/\mathrm{poly}(\lambda)$. Moreover, the latter is true both for indistinguishability and unpredictability games.

We now construct a PPT adversary A' that breaks (ℓ, q)-bounded leakage resilience of Π with the same advantage. A description of A' follows:

- At the onset, A' receives the public value Y and forwards it to A.
- Next, for each round $j \in [q]$, the reduction simulates the queries made by A as described below:
  - Upon input the leakage query $f_j \in \mathcal{F}(X_j)$, the reduction A' runs the simulator $\mathsf{Sim}_1(f_j)$ of Definition 2 in order to get an ℓ-bounded leakage query $g_j \in \mathcal{G}(X_j)$ that is forwarded to the challenger. Hence, the reduction A' returns to A the output $\tilde{Z}_j$ of the simulator Sim_2 on input $g_j(X_j)$.
  - Upon input any further query, this query is passed to the challenger and the corresponding output is returned to A.
- Finally, A' outputs whatever A outputs.

For the analysis, we note that the reduction is perfect in the sense that it perfectly simulates the view of A in experiment $\mathsf{Hyb}_q(\lambda)$. Thus, A' has the same advantage of A. To finish the proof it suffices to note that the running time of A' is dominated by the running time of A plus the time required for running the simulator $\mathsf{Sim}_2(\mathsf{Sim}_1(f_j))$ for each $j \in [q]$. Since the simulator runs in time $t_{\mathrm{sim}} = \mathrm{poly}(\lambda)$, the reduction runs in polynomial time. □

By combining Theorem 10 with Theorem 3 and Theorem 8, setting $\alpha, \delta = \mathrm{negl}(\lambda)$, we obtain cryptographic primitives with game-based noisy-leakage resilience in the practically-relevant settings in which the leakage outputs are either Gaussian-noisy of Lipschitz-noisy.

Corollary 4. *Let $\mathcal{F} = \{f : \mathcal{X} \to \mathcal{Z}\}$ be the family of leakage functions specified as $f(X) = \mathsf{d}(X) + \mathcal{N} = Z$, where $\mathsf{d}(X)$ is polynomial-time computable and supported on an interval $\mathcal{I}$ of length $w_{\mathsf{d}} = \mathrm{poly}(\lambda)$ and $\mathcal{N} = \mathcal{N}(0, \sigma^2)$ is independent of X and follows a centered Gaussian distribution with variance $\sigma^2 = \mathrm{poly}(\lambda)$. Then, assuming that i.i.d. samples from $\mathcal{N}$ and the uniform distribution on $\mathcal{I}$ can be sampled in polynomial time, and that the pdf of $\mathcal{N}$ can be computed in polynomial time, any cryptosystem Π that is $(O(\log \lambda), q)$-bounded leakage-resilient is also $(\mathcal{F}, q)$-noisy leakage-resilient.*

Corollary 5. *Let $\mathcal{F} = \{f : \mathcal{X} \to \mathcal{Z}\}$ be the family of leakage functions such that $f(X) = Z$ is any polynomial-time computable $(\mathcal{R}, \gamma, \delta)$-Lipschitz-noisy leakage with $\mathcal{R} = [a, a + D]$, $a, D \in \mathbb{R}$, $D = 2^{\Theta(\lambda)}$, $D\gamma = 2^{\Theta(\lambda)}$ and $\delta = \mathrm{negl}(\lambda)$. Then, assuming that i.i.d. samples from $[0, d]$ and the distribution P_{Z_x} (for every $x \in \mathcal{X}$) can be sampled in polynomial time, and that the uniform distribution over $[a, a + D]$ and the pdfs P_{Z_x} (for every $x \in \mathcal{X}$) can be computed in polynomial-time, any cryptosystem Π that is $(O(\lambda), q)$-bounded leakage-resilient is also $(\mathcal{F}, q)$-noisy leakage-resilient.*

Simulation-based Style. Next, we consider leakage resilience of interactive protocols (rather than cryptosystems) with simulation-based security. The latter basically means that the adversary, besides corrupting parties, can also leak from the state of the honest players. Since the state includes the inputs, it is often[14] impossible to obtain simulation-based security in the standard sense, but the ideal world functionality needs to be weakened to allow the simulator to get partial leakage from the inputs of honest parties. The latter corresponds to the intuition that a secure protocol does not reveal any information beyond the output and the leakage obtained by the adversary. Sometimes, this guarantees is known as *leakage tolerance* [8].

Below, we consider a basic definition of leakage tolerance for interactive protocols. For simplicity, we stick to standalone executions, but our results can be extended to capture universal composition. A multi-party protocol problem is cast by specifying a random process that maps vectors of inputs to vectors of outputs (one for each party). We refer to such a process as a functionality and denote it $\Phi : (\{0,1\}^*)^n \to (\{0,1\}^*)^n$, where n is the number of parties. For every tuple of inputs $(X_1, \ldots, X_n)$, the output vector is $(\Phi_1(X_1, \ldots, X_n), \ldots, \Phi_n(X_1, \ldots, X_n))$. Leakage tolerance can be defined by comparing two experiments:

- In the ideal world experiment, the parties submit their inputs X_i to a trusted party that computes the outputs on their behalf. An honest party receives

[14] For some tasks, e.g. coin tossing, it is still possible to obtain the standard flavor of simulation-based security.

its input X_i for the computation and just directs it to the trusted party, whereas a corrupted party can replace its input with any other value of the same length. The corrupted parties are controlled by a single adversary S that is additionally allowed to specify a single ℓ-bounded leakage function h_i for each honest party, and obtain the output of h_i applied to the input X_i. For some functionalities, it is impossible to achieve fairness, and thus the trusted party first sends the outputs of the corrupted parties to the adversary, and the adversary then decides whether the honest parties should also receive their outputs or an abort symbol $\perp$; this is called security with aborts. Let Φ be a multi-party functionality, let S be a non-uniform PPT machine, and let $\mathcal{I} \subset [n]$ be the set of corrupted parties. Then, the ideal execution of Φ on inputs $(\lambda, X_1, \ldots, X_n)$, auxiliary input $\xi \in \{0,1\}^*$ to S and security parameter $\lambda \in \mathbb{N}$, denoted $\mathsf{Ideal}_{\Phi,\mathsf{S}(\xi),\mathcal{I}}(\lambda, X_1, \ldots, X_n)$, is defined as the output pair of the honest parties and the adversary in the above ideal execution.

– In the real world experiment, the parties interact directly following the instructions of an interactive q-round protocol Π. The adversary A sends all messages in place of the corrupted parties, and may follow an arbitrary polynomial-time strategy. The honest parties follow the instructions of the specified protocol Π. Additionally, the adversary, at each round, is allowed to leak from the entire state $\Sigma_i^{(j)}$ of honest parties via functions $f_i^{(j)} \in \mathcal{F}(\Sigma_i^{(j)})$, where the state $\Sigma_i^{(j)}$ of party i at round j is defined as a tuple $(X_i, R_i^{(1)}, \ldots, R_i^{(j)})$ where $R_i^{(j)} \in \{0,1\}^*$ is the random tape of party i at round j. Let Φ be as above and let Π be a multi-party protocol for computing Φ. Furthermore, let A be a non-uniform PPT machine and let $\mathcal{I}$ be the set of corrupted parties. Then, the real execution of Π on inputs $(\lambda, X_1, \ldots, X_n)$, auxiliary input $\xi \in \{0,1\}^*$ to A and security parameter $\lambda \in \mathbb{N}$, denoted $\mathsf{Real}_{\Pi,\mathsf{A}(\xi),\mathcal{I}}(\lambda, X_1, \ldots, X_n)$, is defined as the output vector of the honest parties and the adversary in the above real execution.

Definition 9 (Leakage tolerance). *Protocol Π is said to be $(\mathcal{F}, \ell)$-noisy leakage-tolerant in the presence of malicious adversaries if for every non-uniform PPT real-world adversary A there exists a non-uniform PPT ideal-world adversary S, such that for every $\mathcal{I} \subset [n]$:*

$$\left\{\mathsf{Real}_{\Pi,\mathsf{A}(\xi),\mathcal{I}}(\lambda, X_1, \ldots, X_n)\right\}_{\lambda\in\mathbb{N}, X_i, \xi\in\{0,1\}^*} \approx_c \left\{\mathsf{Ideal}_{\Phi,\mathsf{S}(\xi),\mathcal{I}}(\lambda, X_1, \ldots, X_n)\right\}_{\lambda\in\mathbb{N}, X_i, \xi\in\{0,1\}^*}.$$

In case $\mathcal{F} = \mathcal{G}$ coincides with the set of ℓ-bounded leakage functions, we simply say that Π is ℓ-bounded leakage-tolerant in the presence of malicious adversaries.

In the case of bounded leakage, leakage-tolerant interactive protocols have been constructed for several functionalities of interest, including coin tossing [11], zero-knowledge [28, 8], secure message transmission, message authentication, commitments, and oblivious transfer [8]. Follow-up work considered more complex scenarios in which multiple protocol executions take place [2], or the adversary

can leak jointly from the inputs of honest parties [10], or where there is a leak-free pre-processing stage and no leakage in the ideal world [9, 7]. For simplicity, we do not consider these extensions here.

The theorem below says that these protocols also tolerate noisy leakage, so long as the family $\mathcal{F}$ is simulatable from ℓ bits of bounded leakage with polynomial-time simulation and negligible error.

Theorem 11. *Let $\mathcal{F}$ be $(\varepsilon_{\text{sim}}, t_{\text{sim}})$-simulatable from ℓ bits of bounded leakage for $t_{\text{sim}} = \text{poly}(\lambda)$ and $\varepsilon_{\text{sim}} = \text{negl}(\lambda)$. Then, for any q-round interactive protocol Π, if Π is ℓ-bounded leakage-tolerant then Π is also $(\mathcal{F}, \ell)$-noisy leakage-tolerant.*

Proof. We consider a sequence of $q + 1$ hybrid experiments[15] $\mathsf{Hyb}^{(j)}(\lambda)$, for $j \in [0, q]$, specified as follows:

- Experiment $\mathsf{Hyb}^{(j)}(\lambda)$ proceeds identically to the original game defining $(\mathcal{F}, \ell)$-noisy leakage tolerance of Π, except for the way it handles leakage queries $f_i^{(j)} \in \mathcal{F}(\Sigma_i^{(j)})$ for $i \in [n] \setminus \mathcal{I}$, where $\Sigma_i^{(j)}$ is the state of the honest parties at the beginning of the j-th round. In particular:
  - If $k \leq j$, the challenger upon input $f_i^{(k)}$ runs the leakage simulator of Definition 2. By Remark 2, we think of such a simulator as being made of two algorithms $(\mathsf{Sim}_1, \mathsf{Sim}_2)$; algorithm Sim_1 takes as input $f_i^{(k)}$ and outputs a single ℓ-bounded leakage query $g_i^{(k)}$, whereas algorithm Sim_2 takes as input $g_i^{(k)}(\Sigma_i^{(k)})$ and produces the simulated leakage $\tilde{Z}_i^{(k)}$ which is then returned to the adversary.
  - If $k > j$, the challenger upon input $f_i^{(k)}$ returns $Z_i^{(k)} = f_i^{(k)}(\Sigma_i^{(k)})$.

Note that $\mathsf{Hyb}^{(0)}(\lambda)$ is identical to the experiment $\mathsf{Real}_{\Pi,\mathsf{A}(\xi),\mathcal{I}}(\lambda, X_1, \ldots, X_n)$ defining noisy leakage tolerance of Π; the lemma below implies that this experiment is statistically close to the last experiment $\mathsf{Hyb}^{(q)}(\lambda)$, in which all leakage queries are answered using the simulator $(\mathsf{Sim}_1, \mathsf{Sim}_2)$ of Definition 2.

Lemma 15. *For each $j \in [q]$, we have that $\{\mathsf{Hyb}^{(j)}(\lambda)\} \approx_s \{\mathsf{Hyb}^{(j-1)}(\lambda)\}$.*

Proof. Fix a $j \in [q]$. The proof uses a hybrid argument. Let $[n] \setminus \mathcal{I}$ be of size $m < n$. For $i \in [0, m]$, define $\mathsf{Hyb}_i^{(j)}(\lambda)$ to be the experiment that is defined identically to $\mathsf{Hyb}^{(j-1)}(\lambda)$, except that leakage queries $f_k^{(j-1)}$ for $k \leq i$ are answered as in $\mathsf{Hyb}^{(j)}(\lambda)$, whereas leakage queries $f_k^{(j-1)}$ for $k > i$ are answered as in $\mathsf{Hyb}^{(j-1)}(\lambda)$. Note that $\mathsf{Hyb}_0^{(j)}(\lambda) = \mathsf{Hyb}^{(j-1)}(\lambda)$ and $\mathsf{Hyb}_m^{(j)}(\lambda) = \mathsf{Hyb}^{(j)}(\lambda)$. We show that for each $i \in [m]$, it holds that $\{\mathsf{Hyb}_i^{(j)}(\lambda)\} \approx_s \{\mathsf{Hyb}_{i-1}^{(j)}(\lambda)\}$. This implies the lemma.

[15] The hybrids are parameterized by the interactive protocol Π, the attacker A with auxiliary input ξ, the set $\mathcal{I}$ of corrupted parties and a sequence of inputs $X_1, \ldots, X_n$. We omit to write these parameters explicitly in order to simplify the notation.

By contradiction, assume that there is an index $i \in [m]$, an unbounded distinguisher D, an attacker A, an auxiliary input ξ, a set $\mathcal{I}$, a sequence of inputs $(X_1, \ldots, X_n)$ and a polynomial $p(\lambda) = \mathrm{poly}(\lambda)$ such that:

$$\left|\Pr\left[\mathsf{D}(\mathsf{Hyb}_i^{(j)}(\lambda)) = 1\right] - \Pr\left[\mathsf{D}(\mathsf{Hyb}_{i-1}^{(j)}(\lambda) = 1)\right]\right| \geq 1/p(\lambda).$$

We build an unbounded distinguisher D' that contradicts Definition 2 w.r.t. a target distribution $\hat{X}$ which is specified in the reduction below.

- At the onset, D' samples $R_1^{(1)}, \ldots, R_n^{(1)} \leftarrow\!\!\$\ \{0,1\}^*$.
- Next, for each $k \in [q]$, D' answers leakage queries $f_l^{(k)}$ from D as follows:
  - If $k < j$, for every $l \in [n] \setminus \mathcal{I}$, the distinguisher D' returns the value $\tilde{Z}_l^{(k)}$ determined by $\mathsf{Sim}_2(g_l^{(k)}(\Sigma_l^{(k)}))$, where $g_l^{(k)}$ is the ℓ-bounded leakage function returned by $\mathsf{Sim}_1(f_l^{(k)})$, and later sets $\Sigma_l^{(k+1)} = \Sigma_l^{(k)}||R_l^{(k+1)}$ for $R_l^{(k+1)} \leftarrow\!\!\$\ \{0,1\}^*$.
  - If $k = j$, the distinguisher D' proceeds as follows for each $l \in [n] \setminus \mathcal{I}$:
    - If $l < i$, it returns the value $\tilde{Z}_l^{(k)}$ determined by $\mathsf{Sim}_2(g_l^{(k)}(\Sigma_l^{(k)}))$, where $g_l^{(k)}$ is the ℓ-bounded leakage function returned by $\mathsf{Sim}_1(f_l^{(k)})$, and later sets $\Sigma_l^{(k+1)} = \Sigma_l^{(k)}||R_l^{(k+1)}$ for $R_l^{(k+1)} \leftarrow\!\!\$\ \{0,1\}^*$.
    - If $l = i$, it forwards $f_l^{(k)}$ to its challenger for target $\hat{X} = \Sigma_i^{(j-1)}||R_i^{(j)}$ and receives back a pair $(\hat{X}, \hat{Z})$. The value $\hat{Z}$ is then returned to D. Furthermore, it sets $\Sigma_i^{(j+1)} = \hat{X}||R_i^{(j+1)}$ for $R_i^{(j+1)} \leftarrow\!\!\$\ \{0,1\}^*$.
    - If $l > i$, it returns the value $Z_l^{(k)} = f_l^{(k)}(\Sigma_l^{(k)})$, and later sets $\Sigma_l^{(k+1)} = \Sigma_l^{(k)}||R_l^{(k+1)}$ for $R_l^{(k+1)} \leftarrow\!\!\$\ \{0,1\}^*$.
  - If $k > j$, for every $l \in [n] \setminus \mathcal{I}$, the distinguisher D' returns $Z_l^{(k)} = f_l^{(k)}(\Sigma_l^{(k)})$, and later sets $\Sigma_l^{(k+1)} = \Sigma_l^{(k)}||R_l^{(k+1)}$ for $R_l^{(k+1)} \leftarrow\!\!\$\ \{0,1\}^*$.
- The remainder of the interaction between D and its challenger (after the leakage) can be perfectly simulated by D' as it knows $\Sigma_l^{(k)}$ for each $l \in [n] \setminus \mathcal{I}$ and $k \in [q]$.

We note that in case $\hat{Z}$ equals the output of $\mathsf{Sim}_2(g_i^{(j)})$, the view of D is identical to that of $\mathsf{Hyb}_i^{(j)}(\lambda)$; on the other hand, in case $\hat{Z}$ equals the output of $f_i^{(j)}(\Sigma_i^{(j)})$, the view of D is identical to that of $\mathsf{Hyb}_{i-1}^{(j)}(\lambda)$. Hence, the statement of the lemma follows by Definition 2 and by the fact that $\varepsilon_{\mathrm{sim}} = \mathrm{negl}(\lambda)$. □

To conclude the proof, it suffices to observe that experiment $\mathsf{Hyb}^{(q)}(\lambda)$ can be thought of as a special case of the real experiment defining ℓ-bounded leakage tolerance of Π, where the attacker A post-processes each bounded leakage query output $g_i^{(j)}(\Sigma_i^{(j)})$ using algorithm Sim_2. By the assumption that Π is ℓ-bounded leakage tolerant, it follows that for every non-uniform PPT adversary A there exists a non-uniform PPT ideal-world adversary S, such that for every $\mathcal{I} \subset [n]$:

$$\{\mathsf{Hyb}^{(q)}(\lambda)\}_{\lambda \in \mathbb{N}, X_i, \xi \in \{0,1\}^*} \approx_c \{\mathsf{Ideal}_{\Phi, \mathsf{S}(\xi), \mathcal{I}}(\lambda, X_1, \ldots, X_n)\}_{\lambda \in \mathbb{N}, X_i, \xi \in \{0,1\}^*}.$$

This finishes the proof. □

By combining Theorem 11 with Theorem 3 and Theorem 8, setting $\alpha, \delta = \mathrm{negl}(\lambda)$, we obtain interactive protocols with simulation-based noisy-leakage tolerance in the practically-relevant settings in which the leakage outputs are either Gaussian-noisy of Lipschitz-noisy.

Corollary 6. *Let $\mathcal{F} = \{f : \mathcal{X} \to \mathcal{Z}\}$ be the family of leakage functions specified as $f(X) = \mathsf{d}(X) + \mathcal{N} = Z$, where $\mathsf{d}(X)$ is polynomial-time computable and supported on an interval $\mathcal{I}$ of length $w_{\mathsf{d}} = \mathrm{poly}(\lambda)$ and $\mathcal{N} = \mathcal{N}(0, \sigma^2)$ is independent of X and follows a centered Gaussian distribution with variance $\sigma^2 = \mathrm{poly}(\lambda)$. Then, assuming that i.i.d. samples from $\mathcal{N}$ and the uniform distribution on $\mathcal{I}$ can be sampled in polynomial time, and that the pdf of $\mathcal{N}$ can be computed in polynomial time, any interactive protocol Π that is $(O(\log \lambda), q)$-bounded leakage-tolerant is also $(\mathcal{F}, q)$-noisy leakage-tolerant.*

Corollary 7. *Let $\mathcal{F} = \{f : \mathcal{X} \to \mathcal{Z}\}$ be the family of leakage functions such that $f(X) = Z$ is any polynomial-time computable $(\mathcal{R}, \gamma, \delta)$-Lipschitz-noisy leakage with $\mathcal{R} = [a, a + D]$, $a, D \in \mathbb{R}$, $D = 2^{\Theta(\lambda)}$, $D\gamma = 2^{\Theta(\lambda)}$ and $\delta = \mathrm{negl}(\lambda)$. Then, assuming that i.i.d. samples from $[0, d]$ and the distribution P_{Z_x} (for every $x \in \mathcal{X}$) can be sampled in polynomial time, and that the uniform distribution over $[a, a + D]$ and the pdfs P_{Z_x} (for every $x \in \mathcal{X}$) can be computed in polynomial-time, any interactive protocol Π that is $(O(\lambda), q)$-bounded leakage-tolerant is also $(\mathcal{F}, q)$-noisy leakage-tolerant.*